

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO



**FEUP**

# **Context Awareness in an Adaptive Personal Health Assistant**

**Carlos Eduardo Teixeira Tavares**

Dissertação

Mestrado Integrado em Engenharia Informática e Computação

Orientador: Doutor Ademar Manuel Teixeira de Aguiar Professor Auxiliar do  
Departamento de Engenharia Informática da Faculdade de Engenharia da Universidade  
do Porto

Co-orientador: Engenheiro Artur Jorge da Silva Rocha

Julho de 2008



# **Context Awareness in an Adaptive Personal Health Assistant**

**Carlos Eduardo Teixeira Tavares**

Dissertação

Mestrado Integrado em Engenharia Informática e Computação

Aprovado em provas públicas pelo júri:

Presidente: Ana Paula Cunha da Rocha Professor Auxiliar do Departamento de Engenharia Informática da Faculdade de Engenharia da Universidade do Porto

---

Arguente: António Manuel Ferreira Rito da Silva Professor Associado do Departamento de Engenharia Informática do Instituto Superior Técnico

Vogal: Ademar Manuel Teixeira de Aguiar Professor Auxiliar do Departamento de Engenharia Informática da Faculdade de Engenharia da Universidade do Porto

31 de Julho de 2008

# Resumo

Com o aparecimento de plataformas que começam a tornar a visão da computação ubíqua possível, surgem os primeiros sistemas ubíquos, que se caracterizam pela sua invisibilidade, autonomia e conectividade. Uma das aplicações, destes sistemas é a sua utilização como dispositivos pessoais de saúde, abrindo novas possibilidades, por exemplo na telemedicina para monitorização. É possível construir sistemas, móveis, sempre activos, autónomos que permitem uma monitorização médica permanente onde quer que os utilizadores estejam.

Existem já vários sistemas que utilizam conceitos de computação ubíqua, no entanto, nenhum é adaptável às especificidades dos pacientes, ou tem uma abordagem geral da monitorização do estado de saúde do paciente, permitindo apenas o diagnóstico de patologias específicas. O projecto *CAALYX* tem objectivo a construção de um destes sistemas, que utilizam conceitos de computação ubíqua para a monitorização médica, e está a ser desenvolvido pelo INESC Porto. O trabalho realizado nesta dissertação foi realizado no âmbito deste projecto.

Nesta dissertação, é proposta uma solução integrada de um sistema ubíquo, capaz de monitorizar permanentemente um utilizador, utilizando informação derivada de sensores médicos dispostos no corpo do utilizador e de realizar diagnósticos em tempo útil, utilizando conhecimento médico.

Para resolver este problema, que é um caso específico de inferência em contextos multi-sensor, foi escolhida uma abordagem baseada na interpretação de uma linguagem imperativa, que determina como devem ser recolhidos os dados dos sensores, por que ordem e que decisões devem ser tomadas. Em determinadas situações, as recolhas de dados poderão ser realizadas, apenas se determinadas condições forem verificadas. Esta linguagem foi criada no âmbito deste trabalho. A solução obtida é uma solução completa, visto que permite que nenhuma situação de emergência seja perdida, e poupa recursos em relação a uma solução que monitorize os sensores permanentemente.

A arquitectura implementada segue o estilo *blackboard* e foi implementada utilizando uma plataforma móvel J2ME, que integra com *Web Services* para obter dinamicamente todo o conhecimento médico necessário e com os sensores médicos utilizando *Bluetooth*, de modo a obter todos os dados biomédicos.

Foram realizados testes das partes do sistema em separado, incluindo os testes oficiais da primeira fase do projecto e casos de estudo de integração da linguagem criada.

Como conclusão final, pode afirmar-se que é possível a existência de um sistema ubíquo, com capacidades de diagnóstico baseado em conhecimento e este conhecimento, pode ser integrado dinamicamente. No entanto, os testes realizados não permitem comprovar o funcionamento do sistema como um todo.

# Abstract

With the appearance of hardware that begins to make possible the vision of ubiquitous computing, new systems with new approaches on usability, autonomy and connectivity are starting to appear. These new systems, can be applied to personal health devices. This opens new possibilities, for instance, in telemedicine. By this way is possible to build systems that are mobile, always active and autonomous, allowing a permanent medical monitoring despite the user's location.

There are already several systems that use concepts of ubiquitous computation to accomplish the purpose of medical monitoring of the user's health status, however, none of these are adaptable to the user. They are not also capable of making a general medical diagnosis, because they are restricted to some pathologies. CAALYX project is one of these projects that intends to explore ubiquitous and context awareness in personal health systems.

The work carried out in this dissertation, was realized within CAALYX project. In this dissertation, is proposed a integrated solution of a ubiquitous system based on a mobile platform that is capable of monitoring permanently a user using information from medical sensors worn by the user himself and also, that is also capable of executing a general diagnostic using medical knowledge.

To solve this issue, which is itself a specific case of a reasoning problem for multi-sensor contexts, it was chosen an approach based on an imperative language. This language sets the way of collecting sensor's data, the order of this collection, and what decisions should be taken based on the available data. A conditional collection that depends on decisions that were taken based on other sensors, is also allowed. This language was also created within the work done in this dissertation. It is considered that this solution is complete and resource-saving, when compared to intense and permanent analysis approaches.

The implemented architecture follows a *blackboard* style, that was implemented in a J2ME mobile platform. This system integrates with other systems and sensors, using Web Services technology and Bluetooth technology, respectively.

Several tests were carried out, to the several parts of the system. This, included the CAALYX official *trials*, several case studies and the integration with medical knowledge.

As a final conclusion, it can be said that is possible to have a system with diagnostic capabilities that integrate knowledge with external sources. However, the tests made, don't allow to that the system works completely in a real environment.

# Agradecimentos

Gostava de agradecer ao orientador da FEUP, o Professor Ademar Aguiar, pelo interesse que sempre demonstrou por este projecto e pela grandíssima ajuda que me deu, na escrita desta dissertação. Gostava de agradecer também ao co-orientador, Engenheiro Artur Rocha do INESC Porto, pela atenção sempre demonstrada na resolução dos problemas deste projecto e pela liberdade que me deu. Gostava de agradecer ao INESC Porto pela forma profissional como sempre me tratou. Aos colegas do INESC Porto, seguem também os meus agradecimentos, pelo ambiente de trabalho de todos os dias e, por todo o voluntarismo, desde que a minha colaboração no projecto CAALYX começou. Gostava de agradecer também, ao consórcio do projecto CAALYX, por toda a aprendizagem que já me proporcionou. Gostava de agradecer aos meus amigos e família, que tornaram tudo isto tão mais interessante, em particular aos meus pais, por me terem dado tudo! Sem eles não chegaria aqui...

Carlos Eduardo Teixeira Tavares

# Conteúdo

|  |            |
|--|------------|
| <b>Resumo</b>  | <b>i</b>   |
| <b>Abstract</b>  | <b>ii</b>  |
| <b>Agradecimentos</b>  | <b>iii</b> |
| <b>1 Introdução</b>  | <b>1</b>   |
| 1.1 Sensibilidade ao Contexto em Sistemas de Saúde Pessoal . . . . .             | 2          |
| 1.2 Diagnóstico Adaptável em Sistemas Pessoais de Saúde . . . . .                | 2          |
| 1.3 Objectivos do Trabalho . . . . .   | 3          |
| 1.4 Organização da Dissertação . . . . .   | 4          |
| <b>2 Computação Ubíqua em Sistemas Pessoais de Telemedicina</b>                  | <b>5</b>   |
| 2.1 Computação Ubíqua . . . . .  | 5          |
| 2.1.1 Questões em Aberto . . . . .   | 6          |
| 2.1.2 Investigação . . . . .   | 8          |
| 2.1.3 Plataformas Tecnológicas . . . . .   | 8          |
| 2.1.3.1 Dispositivos Móveis . . . . .  | 9          |
| 2.1.3.2 Dispositivos Vestíveis e Tecidos Inteligentes . . . . .                  | 10         |
| 2.1.3.3 Identificação por Rádio Frequência . . . . .                             | 10         |
| 2.2 Sensibilidade ao Contexto . . . . .  | 11         |
| 2.2.1 Localização . . . . .  | 13         |
| 2.2.2 Contextos Visuais . . . . .  | 14         |
| 2.2.3 Contextos Multi-Sensor . . . . .   | 14         |
| 2.3 Sistemas de Saúde Pessoal . . . . .  | 15         |
| 2.3.1 Sensores . . . . .   | 15         |
| 2.3.2 Telemedicina . . . . .   | 15         |
| 2.3.3 Sistemas Ubíquos de Telemedicina . . . . .                                 | 17         |
| 2.3.3.1 Health Service 24, Personal Health Monitoring, Mo-<br>biHealth . . . . . | 17         |
| 2.3.3.2 Vital Jacket, LifeShirt . . . . .  | 17         |
| 2.3.4 Sistemas Ubíquos com Sensibilidade ao Contexto na Telemedicina             | 17         |
| 2.3.4.1 Emerge e DICOEMS . . . . .   | 18         |
| 2.3.4.2 AMON . . . . .   | 18         |
| 2.3.4.3 AUBADE . . . . .   | 18         |
| 2.3.5 Sistemas Sensíveis ao Contexto . . . . .                                   | 19         |
| 2.3.5.1 INTREPID . . . . .   | 19         |

## CONTEÚDO

|          |   |           |
|----------|---|-----------|
| 2.3.5.2  | MYHEART . . . . .   | 19        |
| 2.3.6    | Outros Sistemas Ubíquos . . . . .                                     | 20        |
| 2.3.6.1  | Medical Advisor . . . . .   | 20        |
| 2.3.6.2  | Projecto Code Blue . . . . .  | 20        |
| 2.3.7    | Diagnóstico nos Sistemas Pessoais de Saúde . . . . .                  | 20        |
| <b>3</b> | <b>Diagnóstico em Sistemas Sensíveis ao Contexto</b>                  | <b>22</b> |
| 3.1      | Foco da Dissertação . . . . .   | 23        |
| 3.2      | Trabalho relacionado . . . . .  | 24        |
| 3.2.1    | Sistemas Pessoais de Telemedicina . . . . .                           | 24        |
| 3.2.2    | Sistemas Ubíquos de Telemedicina . . . . .                            | 24        |
| 3.2.3    | Sistemas Ubíquos Médicos Sensíveis ao Contexto . . . . .              | 25        |
| 3.3      | Tese . . . . .  | 25        |
| 3.3.1    | Sistema de Saúde Pessoal Ubíquo . . . . .                             | 25        |
| 3.3.2    | Sensível ao Contexto e com capacidade de Diagnóstico Médico . . . . . | 26        |
| 3.3.3    | Adaptabilidade dos Processos de Decisão . . . . .                     | 26        |
| 3.4      | Resultados Esperados e Métodos de Validação . . . . .                 | 26        |
| <b>4</b> | <b>Sistema Pessoal de Saúde com Capacidades de Diagnóstico</b>        | <b>28</b> |
| 4.1      | Especificação do Sistema Ubíquo . . . . .                             | 29        |
| 4.1.1    | Visão Geral . . . . .   | 29        |
| 4.1.2    | Interacção . . . . .  | 30        |
| 4.1.3    | Sensibilidade ao contexto . . . . .                                   | 30        |
| 4.1.4    | Gestão da Rede de Sensores . . . . .                                  | 31        |
| 4.1.5    | Gestão da Informação do Utilizador . . . . .                          | 32        |
| 4.2      | Sensibilidade ao Contexto . . . . .                                   | 33        |
| 4.2.1    | Contextos Utilizados . . . . .  | 33        |
| 4.2.2    | Identificação do Contexto de Diagnóstico . . . . .                    | 33        |
| 4.3      | Arquitectura . . . . .  | 35        |
| 4.3.1    | Requisitos . . . . .  | 35        |
| 4.3.2    | Arquitecturas semelhantes . . . . .                                   | 35        |
| 4.3.2.1  | Arquitecturas de Agentes . . . . .                                    | 35        |
| 4.3.2.2  | Arquitecturas de Componentes . . . . .                                | 36        |
| 4.3.2.3  | Middleware . . . . .  | 36        |
| 4.3.3    | Integração de Sensores através da <i>Web</i> . . . . .                | 37        |
| 4.3.3.1  | Observações e Medidas . . . . .                                       | 37        |
| 4.3.3.2  | Linguagem de Modelação dos Dados dos Sensores . . . . .               | 38        |
| 4.3.4    | Aplicação dos standards <i>OGC</i> . . . . .                          | 38        |
| 4.3.4.1  | Padrões de Observação . . . . .                                       | 39        |
| 4.3.4.2  | Observações . . . . .   | 39        |
| 4.3.4.3  | Limitações encontradas . . . . .                                      | 40        |
| 4.3.5    | Arquitectura Lógica . . . . .   | 40        |
| 4.3.5.1  | Análises de Informação Médica . . . . .                               | 41        |
| 4.3.5.2  | Rede de Sensores . . . . .  | 41        |
| 4.3.5.3  | Repositório . . . . .   | 42        |
| 4.3.5.4  | Interpretador de Padrões de Observação . . . . .                      | 43        |
| 4.3.5.5  | Linguagem de Observação . . . . .                                     | 45        |

## CONTEÚDO

|          |  |           |
|----------|--|-----------|
| 4.3.5.6  | Interfaces com Sistemas Externos . . . . .                           | 48        |
| 4.4      | Arquitectura Física . . . . .  | 51        |
| 4.4.1    | Integração . . . . .   | 51        |
| 4.4.1.1  | Integração com o Sítio Web . . . . .                                 | 51        |
| 4.4.1.2  | Conectividade com os Sensores . . . . .                              | 51        |
| 4.4.2    | Aplicação Móvel . . . . .  | 52        |
| 4.4.3    | Java para Dispositivos Móveis . . . . .                              | 52        |
| 4.5      | Pormenores de Implementação . . . . .                                | 53        |
| 4.5.1    | Implementação da Comunicação Assíncrona entre Componentes . . . . .  | 53        |
| 4.5.2    | Persistência dos Dados . . . . .                                     | 54        |
| 4.5.3    | Implementação do Interpretador de Observações . . . . .              | 54        |
| 4.5.3.1  | Obtenção de Amostras . . . . .                                       | 55        |
| 4.5.3.2  | Dimensão temporal . . . . .  | 56        |
| 4.5.4    | Localização . . . . .  | 56        |
| <b>5</b> | <b>Experimentação</b>  | <b>58</b> |
| 5.1      | Integração com Sensores . . . . .                                    | 58        |
| 5.1.1    | Desenvolvimento . . . . .  | 58        |
| 5.1.2    | Funcionalidades de Gestão da Rede de Sensores . . . . .              | 59        |
| 5.1.3    | Integração dos Processos de Diagnóstico . . . . .                    | 60        |
| 5.1.4    | Integração com Sítio Web e Integração da Informação Médica . . . . . | 61        |
| 5.1.5    | Integração com Geoposicionamento . . . . .                           | 63        |
| 5.1.6    | Integração Longitudinal com Diagnóstico . . . . .                    | 63        |
| 5.2      | Ensaio da Primeira Iteração do Projecto CAALYX . . . . .             | 64        |
| 5.2.1    | Configuração . . . . .   | 64        |
| 5.2.2    | Descrição do Ensaio . . . . .  | 65        |
| 5.2.3    | Resultados Obtidos . . . . .   | 65        |
| 5.2.3.1  | Resultados Qualitativos . . . . .                                    | 65        |
| 5.3      | Discussão dos Resultados . . . . .                                   | 66        |
| 5.3.1    | Diagnóstico Adaptável . . . . .                                      | 66        |
| 5.3.2    | Linguagem de Diagnóstico . . . . .                                   | 66        |
| 5.3.3    | Arquitectura utilizada . . . . .                                     | 67        |
| 5.3.4    | Tecnologias utilizadas . . . . .                                     | 67        |
| 5.4      | Discussão dos Resultados Obtidos . . . . .                           | 67        |
| <b>6</b> | <b>Conclusões e Trabalho Futuro</b>                                  | <b>69</b> |
| 6.1      | Sumário das contribuições . . . . .                                  | 69        |
| 6.2      | Trabalho Futuro . . . . .  | 70        |
| 6.2.1    | Linguagem das Observações . . . . .                                  | 70        |
| 6.2.2    | Interoperabilidade . . . . .   | 70        |
| 6.2.3    | Implementar arquitectura de <i>Plugins</i> . . . . .                 | 70        |
|          | <b>Referências</b>   | <b>74</b> |

# Lista de Figuras

|      |   |    |
|------|---|----|
| 4.1  | Visão Geral da Solução . . . . .  | 28 |
| 4.2  | Visão Geral dos Casos de Uso . . . . .                                      | 29 |
| 4.3  | Casos de Uso do pacote de Interacção . . . . .                              | 30 |
| 4.4  | Casos de Uso da Sensibilidade ao Contexto . . . . .                         | 31 |
| 4.5  | Casos de Uso do Pacote de Gestão de Sensores . . . . .                      | 31 |
| 4.6  | Casos de Uso do Pacote de Gestão da Informação Médica . . . . .             | 32 |
| 4.7  | Pilha de Contextos Utilizada . . . . .                                      | 33 |
| 4.8  | Exemplo do funcionamento do Motor de Raciocínio . . . . .                   | 35 |
| 4.9  | Arquitectura Geral - Estilo <i>Blackboard</i> . . . . .                     | 41 |
| 4.10 | Funcionamento da Rede de Sensores . . . . .                                 | 42 |
| 4.11 | Modelo de Dados do Repositório . . . . .                                    | 43 |
| 4.12 | Funcionamento do Interpretador de Padrões de Observação . . . . .           | 44 |
| 4.13 | Modelo de Dados dos Padrões de Observação . . . . .                         | 45 |
| 4.14 | Visão Geral das instruções da linguagem dos Padrões de Observação . . . . . | 46 |
| 4.15 | <i>Schema</i> da instrução de obtenção de um conjunto de amostras . . . . . | 46 |
| 4.16 | <i>Schema</i> da instrução de realização de uma análise . . . . .           | 47 |
| 4.17 | <i>Schema</i> da instrução de aplicação de uma condição . . . . .           | 47 |
| 4.18 | <i>Schema</i> da instrução de salto condicional . . . . .                   | 47 |
| 4.19 | <i>Schema</i> da instrução de conclusão . . . . .                           | 48 |
| 4.20 | Interfaces entre os componentes internos e o servidor externo . . . . .     | 49 |
| 4.21 | <i>Schema</i> de alerta com observação . . . . .                            | 49 |
| 4.22 | <i>Schema</i> de uma observação . . . . .                                   | 50 |
| 4.23 | Arquitectura Física da Aplicação Móvel . . . . .                            | 52 |
| 4.24 | Implementação do Padrão <i>Observer</i> . . . . .                           | 54 |
| 4.25 | Estrutura estática do interpretador . . . . .                               | 55 |
| 4.26 | Diagrama de Sequência para obter Medidas . . . . .                          | 56 |
| 5.1  | Funcionalidades de Registo de Sensores . . . . .                            | 59 |
| 5.2  | Execução da leitura dos sensores . . . . .                                  | 60 |
| 5.3  | Thresholds para os sinais vitais . . . . .                                  | 60 |
| 5.4  | Execução da leitura dos sensores . . . . .                                  | 61 |
| 5.5  | Integração dos thresholds . . . . .   | 62 |
| 5.6  | Envio de uma Observação . . . . .   | 62 |
| 5.7  | Envio de electrocardiograma . . . . .                                       | 63 |

# Capítulo 1

## Introdução

A computação ubíqua começa a ser uma realidade, devido à proliferação de plataformas tecnológicas como dispositivos móveis, tecidos e objectos inteligentes. Na visão de *Weiser*, os computadores são “calmos”, não necessitando de qualquer interacção por parte do utilizador e ao mesmo tempo, estas plataformas são uma evolução em relação aos computadores pessoais, necessitando de menos interacção, sendo mesmo completamente desnecessária em alguns casos. Neste âmbito, a sensibilidade ao contexto é a capacidade por parte um sistema ubíquo, de utilizar a informação do mundo externo ao sistema, para melhor adaptar o seu comportamento às necessidades do utilizador, tornando-se mais autónomo. Uma das áreas que mais tem beneficiado com a proliferação deste sistemas é a área médica, nos dispositivos de saúde pessoal e na telemedicina. A aplicação destes conceitos, resulta por exemplo, na criação de sistemas completamente autónomos, inobstrusivos e independentes da localização, destinados a monitorizar indivíduos remotamente e em tempo real.

Existem já bastantes sistemas de saúde pessoal com funcionalidades de telemedicina, que utilizam os conceitos de ubiquidade e de sensibilidade ao contexto. Se estes sistemas, ao aproveitarem estes conceitos, dispuserem de capacidades de diagnóstico geral e forem adaptáveis para cada utilizador, podem melhorar significativamente a monitorização médica, tornando-a mais escalável, mais eficiente e mais barata, em vários aspectos sócio-técnicos.

O problema abordado nesta dissertação, consiste na criação e desenvolvimento de um sistema, baseado numa plataforma ubíqua, com capacidades de diagnóstico utilizando uma estratégia que permita que este seja adaptável.

Este capítulo, pretende introduzir todo o trabalho realizado na dissertação. O capítulo começa por fornecer o *background* necessário às áreas relevantes para este trabalho, nomeadamente, telemedicina, computação ubíqua e sensibilidade ao contexto. Seguidamente, são fornecidos a abordagem e o plano através da qual se cumprirão os objectivos propostos, bem como os resultados esperados.

## 1.1 Sensibilidade ao Contexto em Sistemas de Saúde Pessoal

A sensibilidade ao contexto em sistemas de saúde pessoal, promete trazer autonomia e capacidade de decisão aos dispositivos ubíquos, já de si móveis e invisíveis. Esta, abarca todos os conceitos e problemas relacionados com a recolha, filtragem e inferência de informação, a partir de dados de fontes de informação com semântica pouco significativa. Existem sistemas que utilizam técnicas de sensibilidade ao contexto sofisticadas, para inferir situações complexas a partir de sinais vitais básicos, tais como emoções, ou actividades físicas pouco ou fáceis de distinguir, como a mastigação. Esta área, está ser fortemente investigada e alvo de investimentos por parte de diversas entidades.

O trabalho exposto nesta dissertação, enquadrou-se no projecto “Complete Ambient Assisted Live Experiment” (*CAALYX*). É um projecto de investigação em âmbito europeu, financiado pela Comissão Europeia (CE), ao abrigo do Sexto Programa Quadro (FP6). Este projecto é desenvolvido em consórcio com 8 parceiros, de 6 países diferentes, orçamentado em 3 milhões de euros, com um financiamento global de 1,8 milhões de euros.

Este projecto, tem como objectivos a criação de tecnologia assistida endereçada à população idosa, de modo a aumentar a sua independência e autoconfiança. Pretende-se que isto seja conseguido através da criação de um ambiente assistido, que acompanhe o idoso nos vários ambientes da sua vida diária: o seu ambiente de casa e o ambiente externo. As funcionalidades do sistema, globalmente, são de um sistema de telemedicina que permita aos profissionais de saúde monitorizar o idoso remotamente e inobstrusivamente, sendo estas orientadas para detectar situações de emergência. Entre os vários componentes deste ambiente, o protótipo criado nesta dissertação é o sistema responsável pela monitorização quando o utilizador se encontra em ambiente externo. Este projecto foi realizado no INESC Porto.

## 1.2 Diagnóstico Adaptável em Sistemas Pessoais de Saúde

A existência de capacidades de diagnóstico nos sistemas pessoais de saúde, poderia melhorar muito a autonomia destes, o que possibilitaria sistemas sócio-técnicos de monitorização médica, mais escaláveis, mais eficientes e como tal, mais baratos. Outra melhoria, seria consequência da adaptabilidade do diagnóstico para cada um dos indivíduos, visto que o sistema poderia ser calibrado para cada um dos utilizadores, melhorando os custos de manutenção e evolução do sistema, bem como a sua eficiência. Verificou-se que nenhum sistema até agora desenvolvido tem uma abordagem geral de diagnóstico geral, que seja simultaneamente, adaptável ao utilizador. As soluções encontradas, ou se cingem a patologias específicas ou não são adaptáveis ao utilizador. Isto

reduz a aplicabilidade destes sistemas, porque por exemplo, se os sistemas não têm capacidades de diagnóstico geral, não podem monitorizar todas as patologias fundamentais num cenário real e, se não são adaptáveis, torna-se bastante mais difícil configurar os processos de decisão para um utilizador específico, dificultando a melhoria da eficiência do sistema. A sensibilidade ao contexto, identifica problemas aos quais o diagnóstico se pode reduzir. Ao propor soluções para estes problemas, propõe também, soluções potenciais para este problema. O que se pretende nesta dissertação é dar um contributo para a concretização da resolução do problema de diagnóstico aplicado a um sistema de saúde pessoal.

### 1.3 Objectivos do Trabalho

A adaptabilidade do sistema de diagnóstico, concretiza-se no facto de que os processos de decisão existentes no sistema ubíquo podem ser alterados dinâmica e independentemente da localização, para um utilizador específico. Como tal, o que se pretende fazer nesta dissertação é provar a seguinte afirmação:

Os processos de decisão de um sistema de um sistema de saúde pessoal ubíquo, com capacidade de diagnóstico médico e adaptável, podem ser alterados e evoluídos de forma remota.

Para provar esta tese, os resultados esperados residem basicamente, no desenvolvimento de um protótipo que permita provar a afirmação descrita em 1.3.

Este protótipo, deve ser um sistema ubíquo destinado à monitorização médica remota e deve ter capacidades de diagnóstico e esse diagnóstico deve ser adaptável a cada utilizador. Para provar as funcionalidades descritas acima, deve poder integrar os seus processos de decisão com outro sistema de forma remota e durante o seu funcionamento. A informação do ambiente é dada através de sensores médicos dispostos no corpo do utilizador e através de sensores de localização, de modo a garantir a monitorização independente da localização.

A metodologia utilizada para resolver este problema, começa com a revisão teórica das áreas relacionadas. Isto tem como objectivo, a identificação dos problemas similares ao desta dissertação nessas áreas e das soluções mais utilizadas para os resolver. Deste modo, espera-se poder reduzir o problema, ou os seus subproblemas, desta dissertação, aos problemas encontrados nesta revisão teórica e assim utilizar essas soluções, visto que tanto quanto se sabe, nunca foi tentada nenhuma solução destas, num sistema com estas características. Seguidamente, é desenvolvida uma solução para resolver o problema. Considera-se que a solução se divide numa parte conceptual e numa parte de implementação. A resolução termina com a experimentação do protótipo desenvolvido,

através de experiências desenhadas para o efeito. Pretende-se deste modo, verificar os objectivos e a afirmação desta tese.

## **1.4 Organização da Dissertação**

Esta dissertação, está organizada em mais cinco capítulos. O capítulo 2 faz uma revisão de todos as áreas relacionadas com este sistema, nomeadamente a computação ubíqua, sensibilidade ao contexto, sistemas de saúde pessoais com funcionalidades de telemedicina e a aplicação da sensibilidade ao contexto e computação ubíqua em sistemas de saúde pessoal. O capítulo 3 apresenta o problema desta dissertação e no capítulo seguinte, (capítulo 4), é apresentada a solução proposta para resolver o problema levantado na secção anterior. No capítulo 5 , são descritas as experiências realizadas e os resultados obtidos do trabalho realizado. No capítulo 6 são discutidos os resultados deste trabalho e são apresentadas as conclusões e orientações futuras para este trabalho.

## Capítulo 2

# Computação Ubíqua em Sistemas Pessoais de Telemedicina

A computação ubíqua, promete trazer benefícios para os sistemas pessoais de telemedicina, como inexistência de interacção, omnipresença, autonomia e independência da localização.

Este capítulo, explora todos os assuntos relacionados com a utilização da sensibilidade ao contexto em sistemas ubíquos, começando na computação ubíqua, passando pela sensibilidade ao contexto e finalmente, demonstrando como estes conceitos já se conjugam, em sistemas de telemedicina já existentes.

### 2.1 Computação Ubíqua

A computação ubíqua, deriva de uma visão atribuída a *Mark Weiser* [wei], na qual os computadores estão disponíveis ao utilizador no mundo real, embebidos nos espaços e ambientes do dia a dia, de forma completamente transparente e inobstrusiva para o utilizador, desaparecendo da sua consciência.

O termo ‘ubíquo’ significa omnipresente, que está ao mesmo tempo em toda a parte. *Definição do Dicionário de Língua Portuguesa*

Necessitam de uma interacção mínima, ou inexistente, com o utilizador e tornam-se antes pro-activos, adivinhando as suas intenções, necessidades e cooperando entre eles de modo a atingir os seus fins.

Os primeiros trabalhos nesta área surgiram em 1993, num grupo de investigação do *XEROX Palo Alto Research Center* [par], liderado por *Mark Weiser*. *Mark Weiser* é o autor de algumas publicações nesta área, [Wei99],[Wei98],[WB97] e é considerado o pai da computação ubíqua, visto ter sido ele o primeiro a usar o termo, em 1988.

Em 1988, a computação ubíqua era uma visão completamente inatingível, porque não havia suporte tecnológico para ela, no entanto, têm-se verificado avanços, que cada vez mais, apontam no sentido de que ela venha a ser uma realidade num futuro próximo. Os avanços mais significativos foram os seguintes:

- Lei de Moore<sup>1</sup>
- Novos materiais empregados na construção de computadores, nomeadamente, silício, silício, entre outros;
- Melhoria da conectividade dos dispositivos, nomeadamente devido ao aparecimento de várias tecnologias de rede sem fios, como *UMTS*, *GSM* e *Wifi*, Infravermelhos e *Bluetooth*.

A relação da computação pervasiva com a computação ubíqua, é relativamente difusa. Em [Sea01] considera-se que a computação ubíqua e pervasiva são sinónimos e esta é a definição mais aceite, no entanto outros autores, consideram que existem diferenças entre a computação ubíqua e a computação pervasiva. As diferenças residem na mobilidade dos dispositivos sendo a computação pervasiva referente a dispositivos embebidos em objectos do dia a dia, não necessariamente móveis e a computação ubíqua refere-se a dispositivos com mais mobilidade.

Finalmente, existe uma área de conhecimento denominada "Ambientes Inteligentes". Esta área agrega as áreas da computação ubíqua e pervasiva, e tenta estudar como estas poderão ser utilizadas para criar ambientes "invisíveis", de tecnologia assistida.

### 2.1.1 Questões em Aberto

Considera-se que a computação ubíqua é uma evolução da computação móvel. Por conseguinte, é também a evolução dos sistemas distribuídos, "herdando" portanto, os desafios dessa área. Em alguns casos, podem-se utilizar as soluções da computação móvel, no entanto noutros casos, os problemas são suficientemente diferentes para necessitarem de abordagens e soluções diferentes.

Os desafios levantados pela computação ubíqua, dividem-se por quatro vectores: desenho da interacção, sensores e contexto, descoberta de recursos e infra-estrutura tecnológica. [SN].

**Desenho da Interacção** A computação ubíqua, muda o paradigma actual da interacção com os computadores, porque se pretende que esta seja reduzida ao mínimo ou mesmo, idealmente, a nenhuma, ao contrário do paradigma actual, no qual os computadores são eminentemente interactivos. Assim, o desenho da interacção torna-se mais complexo,

---

<sup>1</sup>A lei de Moore, afirma que a tecnologia de miniaturização dobra a cada 18 meses

visto que a interacção com os dispositivos tem de ser reduzida a um nível quase subconsciente [WB97].

**Sensibilidade ao Contexto e Pró-actividade** Na computação ubíqua, a sensibilidade ao contexto é realizada recorrendo a sensores, que permitem obter dados sobre o ambiente e recorrendo a análises que permitem identificar contextos. A identificação do contexto, é dos aspectos mais importantes para se atingir a visão do computador pró-activo e autónomo de Weiser. Só se a máquina "percepcionar" o que se está a passar, poderá agir em conformidade e prever as necessidades e intenções do utilizador. Os sensores também tomam um papel importante, visto que só tendo os dados é possível retirar conclusões. Nesta área de conhecimento existem vários problemas associados:

- Fusão de Sensores
- Identificação do Contexto
- Monitorização

A fusão de sensores, abarca os problemas relacionados com o funcionamento da rede de sensores tendo em vista a fiabilidade na obtenção dos dados, incluindo situações em que existam sensores redundantes ou conflituosos, na rede. A identificação dos contextos, é o problema central na sensibilidade ao contexto. Existem várias abordagens para identificar contextos, nomeadamente abordagens dedutivas e abordagens indutivas. As abordagens dedutivas, inferem o contexto a partir dos dados obtidos das várias fontes de informação, conhecendo o modelo do contexto à partida. As abordagens indutivas, tentam aprender o modelo do contexto através, por exemplo, de aprendizagem supervisionada. O supervisionamento pode ser realizado, por exemplo, a partir do *feedback* do utilizador. O problema de monitorização, reside na determinação do método de verificação das condições a verificar nos contextos, por exemplo, qual a frequência em que as condições do contexto devem ser verificadas.

**Descoberta e Utilização de Recursos no Ambiente** A redução do tamanho do dispositivo de computação, reduz necessariamente a sua capacidade de processamento. Torna-se fundamental para os computadores ubíquos, descobrir recursos no ambiente e utilizá-los de modo a recolher informação, actuar sobre o ambiente e partilhar tarefas de processamento. Esta área debruça-se, essencialmente, sobre a interoperabilidade entre os dispositivos e protocolos de descoberta de serviços e dispositivos.

**Infra-Estrutura Tecnológica** Na computação ubíqua, pretende-se que os computadores se tornem invisíveis, não só em termos de interação, mas também fisicamente. O tamanho dos dispositivos, não é no entanto, o único obstáculo à invisibilidade. Uma infra-estrutura tecnológica baseada em componentes ubíquos tem de ser fiável, visto que se falha, a sua “invisibilidade” fica posta em causa. Torna-se então realmente importante, construir aplicações tolerantes à falha, com alto grau de dependência e que não sofram degradações rápidas. Outras questões, prendem-se também com a gestão energética dos dispositivos, visto que quanto menos energia gastar mais usável se torna. [Sea01].

**Privacidade e Segurança** Se um computador ubíquo, precisa de adivinhar a intenção do utilizador, de modo a tornar-se pró-activo e autónomo, com certeza, grande parte da informação que irá recolher e processar, é privada e não pode ser conhecida por terceiros. Torna-se então importante, que não partilhe a informação com outros dispositivos sem que estes sejam confiáveis e ao mesmo tempo, que consiga proteger a informação que recolhe, de terceiros.

### 2.1.2 Investigação

Depois de cerca de uma década de avanços tecnológicos, alguns elementos críticos da computação ubíqua, são agora produtos comerciais com bastante sucesso. Neste momento, existem um conjunto vasto de dispositivos móveis, tecidos e roupa inteligentes, casas inteligentes, objectos inteligentes e outros fenómenos de utilizações de princípios da computação ubíqua, como redes *wireless*. Os primeiros projectos mais conhecidos da computação ubíqua, são o *Oxygen* do *Massachusetts Institute of Technology*, *Endeavour* da *Berkeley University* e *Aura* da *Carnegie Mellon University*[GSSS02].

Neste momento, a computação ubíqua é alvo de intensa investigação, tanto da indústria e comunidade académica, como de instituições governamentais e isto, verifica-se um pouco por todo o mundo. Existem já milhares de projectos relacionados com *software* e *hardware* para sistemas ubíquos, com a criação de novas plataformas e artefactos.

Na Europa, uma das iniciativas mais importantes é o *The Disappearing Computer*[DCI], que é financiada pela Comissão Europeia, e o *Centre for Pervasive Computing* da Dinamarca. Neste centro, existem várias iniciativas relacionadas com a utilização da computação ubíqua na área da saúde.

Comercialmente, existem algumas empresas que também realizaram trabalhos nesta área, nomeadamente, *IBM*, *AT&T*, *Fuji* e *Sony*.

### 2.1.3 Plataformas Tecnológicas

Recentemente, têm aparecido plataformas que começam a permitir a concretização da visão da computação ubíqua. Estas plataformas dividem-se em dispositivos móveis,

*wearable devices*, tecidos inteligentes e muito recentemente, electrónica implantável.

### 2.1.3.1 Dispositivos Móveis

Os dispositivos móveis, são dispositivos computacionais que por definição, são suficientemente pequenos para poderem ser transportados num bolso e podem ser utilizados enquanto o utilizador os segura na mão. São normalmente utilizados para fins pessoais como comunicação e têm recursos mais reduzidos quando comparados com computadores de secretária, ou portáteis. Existem vários tipos destes dispositivos no mercado, tais como:

- *Personal Digital Assistants(PDA)*
- *Smartphones*
- *Pocket PC*
- *Pager*
- *Consolas de Jogos Portáteis*

Os dispositivos que têm mais possibilidades de cumprirem o papel de dispositivos ubíquos, principalmente pelas suas capacidades de conectividade e processamento, são os *Smartphones* e alguns tipos de PDAs. Estes dispositivos, baseiam-se em plataformas de *software* similares aos computadores pessoais, com sistemas operativos e *apis* que os tornam programáveis e que permitem o desenvolvimento de aplicações para os mesmos.

Os *Smartphones*, e alguns tipos de PDAs, estão equipados com uma série de transdutores que se destinam a fins de comunicação, tais como, microfones, auto-falantes, câmaras e monitores, que têm sido aproveitados em alguns projectos de computação ubíqua, como fontes de informação para sensibilidade ao contexto.

Para além destes recursos, estes dispositivos móveis começam a ter suporte para sensores de localização, seja através de módulos de *GPS* integrados nos PDAs e *Smartphones* mais recentes, ou através do cálculo do posicionamento da triangulação das células de difusão da rede móvel. Começam também, a ser disponibilizados outros tipos de sensores nos dispositivos mais recentes, como acelerómetros.

Outro aspecto importante, que torna estes dispositivos privilegiados para a computação ubíqua, é a sua conectividade. Os dispositivos móveis podem utilizar várias redes que lhe dão acesso a diferentes serviços de dados: *GSM*, *UMTS*, *GPRS* e em alguns exemplos recentes, têm-se acesso à tecnologia *WiFi*.

Alguns dispositivos móveis, têm também interfaces rede para tecnologias *wireless* de curto alcance como *IrDa* e *Bluetooth*. Estas tecnologias permitem a integração com outros dispositivos, o que inclui sensores.

Outro dos factos importantes é a possibilidade de troca de mensagens de assíncronas e seguras, utilizando *Short Message Service(SMS)*. Esta tecnologia, é utilizada por milhares de pessoas, todos os dias.

Ao mesmo tempo que tudo isto melhora, a autonomia destes dispositivos diminui, o que se torna um problema de usabilidade. Isto é aliás o principal problema de usabilidade, visto que os dispositivos móveis, podem facilmente ser transportados num bolso. Podem também, ser completamente autónomos, se o *software* desenvolvido for nesse sentido, embora originalmente, não tenham sido projectados para ser completamente autónomos.

### 2.1.3.2 Dispositivos Vestíveis e Tecidos Inteligentes

Dispositivos vestíveis são dispositivos computacionais que, de algum modo, podem ser vestidos.

Estes dispositivos aparecem em relógios inteligentes, pulseiras ou ainda tecidos inteligentes e são já utilizados para vários fins, entre os quais a saúde, sendo essa a sua principal utilização. Ao contrário dos dispositivos móveis, não existem dispositivos de uso geral programáveis que possam ser transformados para um fim específico. Na sua maioria, os dispositivos vestíveis são conceptualizados e desenvolvidos para um único fim. Existem no entanto algumas plataformas, de uso geral tais como a plataforma *Smart-Its*[BG03] e *software* como o sistema operativo *tinyOs*. A plataforma *Smart-Its* consiste em pequenas células de *hardware*, que podem ser vestidas e que são programáveis.

A interacção com este tipo de dispositivos é a ideal, visto que são tão usáveis como uma peça de roupa, o que os enquadra perfeitamente, na visão de *Mark Weiser*.

A conectividade destes dispositivos depende da implementação. Existem sistemas, a utilizarem algumas tecnologias comuns, como *GPRS*, *GSM* e rádio frequência, no entanto, existem outros que utilizam canais de transmissão de dados menos comuns, tais como tecidos condutores, ou mesmo utilizar a pele para transmissão de dados.

Ao mesmo tempo, para além técnicas de alimentação energética mais comuns, como a utilização de baterias, é possível utilizar a energia do utilizador nas suas actividades diárias, melhorando a sua autonomia. Neste aspecto particular, foram experimentadas várias fontes de energia, tais como movimentos dos braços, passos, pressão arterial ou ainda a temperatura corporal [Sta96].

### 2.1.3.3 Identificação por Rádio Frequência

As *rfid tags* são uma tecnologia emergente. São constituídas pequenas etiquetas, com pequenas placas de silício, e são maioritariamente utilizados para identificação. Uma *tag* é um pequeno objecto tem um identificador único que é associado a algo, como uma pessoa, animal ou um objecto. Este objecto é capaz de transmitir o seu identificador por rádio frequência. Uma etiqueta pode funcionar em modo passivo, ou em modo activo. O

modo passivo apenas reflecte sinais, ao passo que as etiquetas em modo activo são capazes de transmitir o seu próprio sinal. Isto significa que no primeiro caso, as etiquetas só transmitem o seu identificador quando inquiridas através de um leitor, ao passo que no segundo caso as células são capazes de transmitir o seu identificador activamente. Devido à existência de células activas, é esperado que esta tecnologia venha a ser utilizada para outros fins que não apenas a identificação. Os tipos de objectos aos quais as tags podem ser associadas são muitos e variados e, vão desde etiquetas autocolantes, ou mesmo comprimidos deglutíveis. [GR05]

## 2.2 Sensibilidade ao Contexto

Na sua definição geral, a palavra contexto significa "conjunto de circunstâncias em que algo acontece ou existe". No domínio da computação ubíqua, esta definição é porém, diferente, porque embora existam inúmeras possibilidades de circunstâncias identificáveis e diferenciáveis por um dispositivo ubíquo, nem todas são relevantes para a interacção entre o utilizador e o computador ubíquo. Como tal, é proposta uma definição formal de contexto:

Contexto é qualquer informação que pode ser utilizada para caracterizar a situação de uma dada entidade. Uma entidade pode ser uma pessoa, um lugar, ou um objecto que seja relevante para a interacção entre o utilizador e a aplicação

Não satisfeitos com a definição proposta, por a acharem ainda demasiado abrangente, em [CK00] identificam-se os tipos de contextos possíveis no âmbito de uma aplicação ubíqua.

- Contexto Computacional (Conectividade à rede, custo das comunicações e largura de banda, outros recursos que estejam na proximidade, etc.)
- Contexto do Utilizador (Perfil do utilizador, localização, pessoas na proximidade, situação social ou emocional, etc.)
- Contexto Físico (Luz, níveis de ruído, temperatura, humidade)
- Contexto Temporal (Hora do dia)

Considera-se ainda, a existência de contextos de alto nível e de contextos de baixo nível. Contextos de alto nível, são contextos que não são mensuráveis através de sensores, mas que podem ser inferidos através de análises complexas, a partir de dados dos mesmos.

Uma aplicação sensível ao contexto, tem a capacidade de identificar a situação em que se encontra e de utilizar o contexto na sua execução. Um dos desafios na investigação

das aplicações sensíveis ao contexto, tem sido identificar os contextos que podem ser utilizados e a forma que permite que estes possam ser utilizados de forma eficiente. Shillit [Sch95] tenta definir os vários tipos de aplicações da sensibilidade ao contexto:

1. Seleção por proximidade. A aplicação utiliza o seu contexto computacional para identificar recursos e enfatizar os objectos mais próximos. Isto tem como objectivo, facilitar a selecção dos recursos pelo utilizador.
2. Reconfiguração automática em contexto. Um processo de adição de novos componentes, de remoção de componentes, ou de alteração de ligações entre componentes baseado em mudanças detectadas no contexto;
3. Informação Contextual e comandos. As aplicações produzem resultados diferentes consoante o contexto;
4. Acções despoletadas pelo contexto. Acções tipo IF-THEN que definem como a aplicação deve actuar em relação ao contexto;

A identificação dos contextos é o problema mais relevante da sensibilidade ao contexto. Existem várias abordagens possíveis para identificar contextos cada vez mais complexos. Estas abordagens, dividem-se maioritariamente em três tipos:

- Antecipação ao Contexto
- Dedução de Contextos
- Aprendizagem de Contextos

A antecipação ao contexto, tenta fundamentalmente, “adivinhar” quando será mais provável que este se altere. A antecipação, pode servir como indicador da mudança de contexto, de modo a activar previamente componentes necessários à mudança de contexto, ou simplesmente para otimizar as frequências de *pooling* de verificação das mudanças de contexto. As técnicas utilizadas são maioritariamente, técnicas de previsão. Estas técnicas, podem ser séries temporais, tais como filtros de *Kallman*, ou técnicas baseadas em amostragem de *Monte Carlo* sequencial.[TLL05]

A aprendizagem do contexto, consiste em “treinar” a aplicação para que esta delimite os contextos a partir dos dados obtidos, através de um processo de classificação como por exemplo o *feedback* do utilizador. Nestas abordagens, é a aplicação que constrói o modelo do contexto, ao invés deste lhe ser fornecido à partida.

A dedução do contexto, é a maneira mais imediata de o obter. Nos contextos de baixo nível, não é necessária a dedução, mas para contextos mais complexos, derivados por vezes de vários sensores, é necessário algum tipo de dedução e análise dos dados dos sensores, existindo também diferentes técnicas para este fim.

### 2.2.1 Localização

Existem muitas e variadas técnicas para a determinação de localização em dispositivos ubíquos sendo este um contexto muito utilizado na computação ubíqua. Algumas destas técnicas estão disponíveis apenas em dispositivos criados para o efeito, outras estão largamente difundidas e estão acessíveis em dispositivos disponíveis comercialmente. A localização é uma fonte de informação com um largo espectro de aplicação, nomeadamente em sistemas de navegação ou em novos serviços baseados em localização.

**Localização Celular** A localização celular está disponível em dispositivos móveis como *Smartphones* e *PDA's* mais recentes, desde que estes tenham acesso às redes móveis. Consiste no cálculo da distância, a pelo menos, três estações de diferentes células transmissoras do sinal móvel. Existem vários algoritmos para o cálculo do geoposicionamento desta forma, baseados na estimativa da atenuação do sinal entre os dispositivos e as antenas de difusão celular.

**Global Geopositioning System** Global Positioning System, é um sistema de geoposicionamento acessível de toda a superfície terrestre, composto por um conjunto de satélites, que fazem *broadcast* do seu próprio geoposicionamento. Um dispositivo pode calcular o seu geoposicionamento, calculando a sua distância a pelo menos três destes satélites e realizando a intersecção das circunferências resultantes, o que lhe permitirá obter um triângulo suficientemente pequeno, para constituir o seu geoposicionamento. Este sistema, pode ser acedido por receptores específicos e é suportado pela Marinha dos Estados Unidos. Outras entidades, estão a preparar alternativas a este sistema, nomeadamente, a União Europeia, China, Rússia e Índia.

**Outras Abordagens** Existem várias soluções para obtenção de localização, que se baseiam fundamentalmente na utilização de radiações electromagnéticas, como ondas de rádio, infravermelhos e ondas ultra-sónicas. Isto é realizado de várias maneiras diferentes, com diferentes graus de eficiência. Estas soluções, não estão na sua maioria, disponíveis comercialmente, porque foram especificamente desenvolvidas em contextos de investigação. O *active badge system* da *Olivetti*, o *Xerox ParcTab* e o *Cyberguide Project* utilizaram soluções de localização baseadas em infravermelhos. O *Personal Shopping Assistant* da *AT&T* utilizaram uma solução de radio frequência pura. O projecto *RADAR* utilizou o sinais rádio das redes WiFi, para calcular a localização. [CK00]

### 2.2.2 Contextos Visuais

Os contextos visuais, são na sua generalidade contextos de alto nível, que são inferenciáveis a partir de fontes visuais e utilizam técnicas de visão computacional para esse fim.

**Detecção de Objectos e Ambientes** As técnicas de visão computacional, podem ser utilizadas para identificar vários parâmetros visuais do utilizador, tendo como tal, uma aplicação bastante diversificada. Um dos contextos que pode ser inferenciado, utilizando técnicas de detecção de contextos visuais, é o contexto de identificação de ambientes e de proximidade de objectos. Cada objecto pode ter associado um padrão visual bem definido, através de uma etiqueta ou outro identificador visual, que permite identificar esses objectos visualmente.

**Detecção de gestos** A detecção de gestos, também pode ser inferenciada a partir de fontes de informação visual. Esta, tem um grande âmbito de aplicação, visto que pode ser utilizada para estabelecer uma interface homem-máquina utilizando gestos e já foi utilizada em alguns projectos, como indicador do contexto emocional do utilizador.

### 2.2.3 Contextos Multi-Sensor

Os contextos multi-sensor, são contextos que dependem da integração de informação de diferentes sensores. Este problema, é um caso específico do problema de integração de informação. As abordagens utilizadas, dividem-se na utilização de técnicas estatísticas e de inteligência artificial, ou técnicas baseadas em conhecimento, como ontologias, ou técnicas que misturam as duas estratégias anteriores.

**Ontologias e Raciocínio Baseado em Lógica** A utilização de ontologias, é bastante frequente para representar conhecimento. No caso específico da sensibilidade ao contexto, as ontologias podem ser utilizadas para representar todas as situações contextuais e as relações lógicas entre elas. Assim, é possível calcular, utilizando raciocínio baseado em lógica, qual o contexto actual do sistema, necessitando para tal, de uma linguagem que permita representar o conhecimento do contexto e de um motor de inferência para o cálculo situacional.

Uma das linguagens mais utilizadas, para representar ontologias é *OWL*. Esta linguagem, foi recentemente escolhida como padrão da W3C para representação de conhecimento na *internet*. É um caso específico da linguagem *XML*. Existem também, várias casos de linguagens para representar ontologias, que não têm um âmbito de aplicação tão generalizado como *OWL*, sendo específicas de determinados domínios.

**Raciocínio Situacional Probabilístico** As técnicas de raciocínio probabilístico, permitem a criação de modelos baseados em probabilidades. Existem várias técnicas possíveis como utilizar redes de *Bayes* ou modelos de *Markov*, ou ainda redes neuronais.

## 2.3 Sistemas de Saúde Pessoal

Os sistemas de saúde pessoal, são uma classe de sistemas que engloba um conjunto de dispositivos electrónicos para uso pessoal, que regra geral, se aplicam à monitorização e análise do estado de saúde de um indivíduo. Recentemente, esta área tem-se desenvolvido muito e começam a existir sistemas baseados em plataformas ubíquas, como dispositivos móveis, dispositivos vestíveis, ou mesmo implantáveis.

### 2.3.1 Sensores

Existem vários fabricantes de sensores médicos portáteis, tais como *Omron Health Care*, *Phillips*[phi], *Eckerd*, *CORSCIENCE*[cor], entre muitos outros. Os sensores médicos são de uma grande variedade e quantidade e aplicam-se a uma grande quantidade de sinais vitais, que vão desde a temperatura, pressão arterial, até sinais vitais mais complexos tais como electroencefalogramas. Estes sensores possuem memória, o que pode permitir a o armazenamento dos seus dados para posterior análise, ou a integração com outros nós de processamento, como sistemas de telemedicina capazes de enviar os dados para profissionais de saúde, que prestam assistência médica. Existem muitos tipos de dispositivos para este fim, que vão desde sensores médicos portáteis, como sensores de oximetria, pressão arterial, electrocardiografos portáteis, até sistemas de telemedicina que podem ser utilizados em casa, que comunicam com serviços centrais, e que permitem o envio de sinais vitais, obtidos de outros sensores. Um tipo especial destes sensores, são os sensores de corpo, que se encontram integrados através de tecidos inteligentes. Este tipo de dispositivos, devido ao tamanho da superfície que se encontra em contacto com o corpo, trazem outro tipo de potencialidades no que diz respeito à sensorização e aos contextos sociais e médicos do utilizador.

### 2.3.2 Telemedicina

A telemedicina, é um termo que se refere ao exercício da medicina à distância, através da utilização de tecnologias de comunicação e informação. Faz parte de uma área denominada Medicina Electrónica, que abarca toda a utilização de tecnologias da informação e comunicação na área médica. Esta área, inclui, por exemplo, os *Personal Health Records* e repositórios de conhecimento médico.

A telemedicina aplica-se a fins tão diversos como a monitorização remota de pacientes, imagiologia médica à distância e recentemente, foi aplicada em procedimentos cirúrgicos realizados à distância.

Existem dois tipos de telemedicina: Telemedicina em tempo real e assíncrona. A telemedicina de tempo real é realizada sincronamente, ou seja, embora realizada à distância, exige a interacção em tempo real de ambos os intervenientes.

Na telemedicina assíncrona a informação é enviada por um dos intervenientes e utilizada pelo outro interveniente em modo *off-line*, não exigindo a presença de ambos os intervenientes em simultâneo.

Existem vários sistemas de telemedicina direccionados para a monitorização médica, que têm diferentes famílias de funcionalidades. Estes sistemas, recolhem dados médicos dos utilizadores e enviam-nos para os profissionais de saúde disponíveis remotamente, quer em tempo real, quer assincronamente.

Alguns destes sistemas, podem efectuar diagnósticos médicos sobre os dados. A monitorização médica, com este tipo de dispositivos, é utilizada para fins tão diversos como monitorização de pacientes em reabilitação, monitorização de pacientes de risco, ou mesmo detecção de crises agudas e emergências.

Existem finalmente, sistemas de telemedicina para uso pessoal. São sistemas que funcionam, por exemplo, a partir de casa e que são capazes de recolher leituras de sinais vitais do utilizador, através da integração com outros sensores médicos.

Alguns dos sistemas, funcionam como simples *hubs*, que recebem dados dos sensores e enviam para servidores na *internet*, onde estes podem ser recebidos e analisados por profissionais de saúde. Por vezes, estes sistemas também dispõem de funcionalidades como agendas de medições a realizar pelo utilizador, botões de alerta e canais áudio ou vídeo, que lhes permitem comunicação directa com os profissionais de saúde. Existem sobretudo soluções comerciais para este tipo de sistemas, como por exemplo, *TeleStation* da *Phillips* e *Remote Nurse* da *Homem Telehealth*. Outros sistemas, são o *CareCompanion Home Monitoring System*[[car](#)], *TeleMedCare System*[[tel](#)] e *HealthBuddy*[[heaa](#)]. O sistema *Health Buddy*, permite a especificação de perguntas ao utilizador de modo identificar sintomas e não apenas sinais.

Outra classe de sistemas de saúde pessoal, são dispositivos que não lêem nenhum sinal vital, mas que dispõem de botões que permitem pedir ajuda em caso de necessidade. Existem várias soluções comerciais neste tipo de sistemas, fornecidas por exemplo, pela *Phillips*.

### 2.3.3 Sistemas Ubíquos de Telemedicina

#### 2.3.3.1 Health Service 24, Personal Health Monitoring, MobiHealth

Os projectos *Health Service 24*[[heab](#)], *Personal Health Monitoring*[[per](#)] e *MobiHealth*[[Ric](#)] baseiam-se em dispositivos móveis e partilham a mesma abordagem para a monitorização. Os dispositivos móveis, lêem permanentemente os dados dos sensores da *BAN* do utilizador e enviam esses dados, também permanentemente, para um sítio na *Internet* onde essa informação pode ser visualizada em tempo real. Neste cenário, o dispositivo móvel é um nó do canal de comunicação, que não desempenha quaisquer funcionalidades de análise, ou filtragem dos dados, deixando a análise e tomada de decisão para os profissionais de saúde que terão acesso à informação no destino. O projecto *Health Service 24*, foi desenvolvido no *Fifth Framework Programme* da Comissão Europeia e utiliza um *PDA*, que envia dados por um canal *UMTS*, permanentemente. O projecto *Personal Health Monitoring System* [[per](#)], enquadra-se também neste tipo de abordagens e foi o primeiro sistema do género. O projecto *MobiHealth* é também um projecto europeu, que tem como particularidade, o facto de enviar a informação médica através de serviços que é capaz de descobrir na rede em que se encontrar no momento, tendo portanto, a capacidade de descobrir serviços no ambiente.

#### 2.3.3.2 Vital Jacket, LifeShirt

O cenário mais simplificado utilizando tecidos inteligentes, consiste em que estes sejam apenas agregadores de sensores. Os dados resultantes destes sensores, podem ser guardados numa memória que pode ser analisada mais tarde por profissionais de saúde, ou podem ser lidos em tempo real por outro dispositivo, como um *Smartphone*.

Um destes projectos é o *Vital Jacket*[[vit](#)], desenvolvido na Universidade de Aveiro. É composto por uma vulgar *t-shirt*, com vários sensores vitais acoplados. Este sistema, permite o armazenamento dos dados recolhidos, ou o envio dos dados para outros dispositivos *wirelessly*.

Do mesmo modo, o *LifeShirt*[[lif](#)] consiste num colete capaz de monitorizar diversos sinais vitais, como batimento cardíaco e número de respirações por minuto. O *LifeShirt* é já um produto comercial e também neste sistema, os dados são registados conjuntamente com a sua *timestamp*, num dispositivo capaz de os armazenar. Os profissionais de saúde consultam os dados presentes nesse dispositivo mais tarde.

### 2.3.4 Sistemas Ubíquos com Sensibilidade ao Contexto na Telemedicina

Uma das abordagens da utilização da computação ubíqua na telemedicina, consiste na delegação de mais "responsabilidade" aos sistemas na decisão, tornando-os capazes de filtrar informação médica que não necessite de atenção por parte dos profissionais

de saúde. Estes sistemas são sensíveis ao contexto, reagindo e inferenciando dos dados obtidos dos sensores.

#### 2.3.4.1 Emerge e DICOEMS

O projecto *EMERGE* [eme] tem como finalidade, a monitorização da população idosa para fins de prevenção, assim como a detecção de estados de saúde que necessitem de atenção médica. Quando são detectados estes estados de saúde, estes são notificados através de um sítio na *Internet*. No processo de notificação, é também fornecido o geoposicionamento do utilizador. Este projecto está a ser desenvolvido no *Sixth Framework Programme* e ainda não existem resultados públicos dele. O projecto *DICOEMS* [dic], segue a mesma abordagem do projecto anterior utilizando vários sensores. É capaz de avaliar a situação clínica do utilizador, de notificar a rede de especialistas e de enviar o geoposicionamento associado ao episódio médico. Ambos os dispositivos são sensíveis ao contexto e capazes de realizar diagnósticos. O projecto *DICOEMS*, do qual existem resultados públicos, utiliza um conjunto de regras predefinido para realizar o diagnóstico.

#### 2.3.4.2 AMON

O projecto AMON Monitoring [Neu02], desenvolvido no *Fifth Framework Programme* da Comissão Europeia, utiliza um dispositivo vestível em forma de pulseira para monitorizar pacientes cardíacos de risco. Utiliza vários sensores para a monitorização em tempo real dos pacientes e um conjunto de algoritmos e regras, para analisar e relacionar os dados dos vários sensores, utilizando como tal, um contexto multi-sensor. Utiliza acelerómetros de dois eixos para detectar contextos de actividade do paciente e relaciona esta informação com as análises cardíacas. Envia os dados, para uma central de monitorização, utilizando *UMTS* ou *GSM* e suporta pedidos assíncronos, enviados a partir da central de monitorização. Os dados enviados são apenas os que resultem da detecção de algo significativo, ou dados pedidos pela central de monitorização. Os sensores, no entanto, têm de ser activados pelo utilizador o que permite concluir, que o sistema não é completamente autónomo.

#### 2.3.4.3 AUBADE

O projecto *AUBADE*[CDKF06], tem uma configuração similar com o projecto *AMON*, visto que funciona com base num dispositivo vestível. Este dispositivo vestível, é capaz de monitorizar em tempo real e de enviar para um servidor central as emoções básicas do utilizador, a partir dos dados recolhidos através de expressões faciais e electrocardiogramas. Utiliza um contexto multi-sensor, em que alguns dos sensores, produzem grandes

quantidades de informação. Está a ser desenvolvido sobre a alçada do *Framework Programme 6*, fundamentalmente para fins de prevenção, acompanhamento e investigação de doenças cerebrais.

## 2.3.5 Sistemas Sensíveis ao Contexto

### 2.3.5.1 INTREPID

O projecto *INTREPID* [int], é outro dos projectos que utiliza um *wearable light device*, desta vez para monitorizar o estado emocional de pacientes fóbicos, na presença das suas fobias. Os sintomas das fobias, são monitorizadas a partir de vários sensores relacionados com as emoções, como pressão arterial, ritmo cardíaco e número de respirações/minuto. A monitorização das fobias, tem como objectivo tornar mais fácil a prescrição de tratamentos aos pacientes e para compreender o seu comportamento na presença das mesmas.

### 2.3.5.2 MYHEART

O projecto *MYHEART*[LSD<sup>+</sup>06], desenvolvido maioritariamente pela *Phillips*, tem como objectivo produzir "roupa inteligente" sensível ao contexto dos sinais vitais e de dar *feedback* ao utilizador sobre as suas acções. O sistema aproveita a grande superfície de contacto com corpo para obter um conjunto de sinais vitais e de análises, que de outro modo, não seriam possíveis. Embora as suas linhas directivas, se orientem no sentido da prevenção e detecção de doenças cardiovasculares, a plataforma tecnológica irá ser utilizada de diferentes maneiras. Exemplos das aplicações deste projecto são programas de treino físico, com *feedback* contínuo com base nas análises sobre dados cardíacos recolhidos continuamente, ou programas de relaxamento durante o sono.

Outra das aplicações, das mais inovadoras deste sistema, é o controlo dietético automático tendo em vista a prevenção de doenças cardiovasculares.

O controlo dietético automático baseia-se na análise dos padrões de alimentação do individuo, [JALT08], [AT08], [AKT07], [RABB07], [ATLS06],[AJT05],[ALO<sup>+</sup>04]. O projecto *MYHEART*, analisa contextos tais como o som da mastigação e a movimentação física dos músculos do peito, de modo a tentar associar estes parâmetros à mastigação de alimentos menos aconselháveis para doenças cardíacas. Utiliza para tal, vários contextos multi-sensor, no entanto, estes contextos não são adaptáveis. São utilizadas na maior parte das vezes, abordagens de *caixa negra*, através de técnicas estatísticas, redes neuronais, séries temporais, etc.

## 2.3.6 Outros Sistemas Ubíquos

### 2.3.6.1 Medical Advisor

Segundo pareceres médicos noventa por cento do diagnóstico, é realizado com base na entrevista com o paciente. Apenas foi encontrado um projecto, que segue esta abordagem. Este projecto, é o *Medical Advisor* [FAB<sup>+</sup>02] da iniciativa *Conversational Health Assistants* da *Universidade de Rochester*. Neste momento, é utilizado um dispositivo móvel e através de técnicas de análise de um contexto de conversação, é tentado o diálogo com o paciente, de modo a realizar uma entrevista médica.

### 2.3.6.2 Projecto Code Blue

O projecto *Code Blue* [SrCL<sup>+</sup>05], desenvolvido pela *Harvard University*, baseia-se na tecnologia *Smart-Its* [BG03]. As células de *Smart-Its* são vestidas pelo utilizador. Estas células são utilizadas para interagir com sensores colocados no corpo do utilizador e raciocinar sobre os dados dos mesmos. Neste projecto, é proposta uma abordagem de raciocínio em rede através da cooperação entre as várias células. O sistema pode enviar os dados para um *PDA/Smartphone*, ou para qualquer outro recurso que esteja ao alcance das células, como por exemplo, um computador pessoal. O sistema pode ser utilizado para fins de detecção, mas também para fins de prevenção. Não são conhecidas no entanto, utilizações reais deste projecto e não é independente da localização, visto que está restringido a um ambiente específico.

## 2.3.7 Diagnóstico nos Sistemas Pessoais de Saúde

O diagnóstico é uma abordagem racional da medicina, baseada na dedução lógica, na qual entram apenas duas entidades, sinais e sintomas. Os sinais, advêm de grandezas médicas mensuráveis, como tensão arterial, batimento cardíaco, electrocardiograma, etc. Os sintomas, são todos os factos médicos que não são mensuráveis e normalmente, estão associados a queixas do paciente, como dor, depressão, ou emoções. O diagnóstico pode também ser influenciado por outros contextos, como a actividade física do utilizador, visto que os padrões de certos sinais vitais, podem ser alterados por essa mesma actividade física. A sensibilidade ao contexto, é utilizada em vários sistemas desenvolvidos em âmbito de investigação, como foi visível nas últimas secções. A utilização dos contextos analisados, resultam em algumas situações, na detecção de determinados padrões e patologias, ou então, na detecção de actividades físicas em tempo real. Não foi encontrado nenhum sistema com uma abordagem de diagnóstico médico geral, limitando-se a determinadas classes de patologias, embora algumas destas aproximações sejam muito interessantes e inovadoras. Outra das possíveis técnicas que poderão ser muito úteis num

sistema de diagnóstico, residem na entrevista com o paciente. Um dos projectos analisados, segue uma abordagem baseada num contexto de conversação, que permitirá a implementação de um sistema com estas capacidades.

## Capítulo 3

# Diagnóstico em Sistemas Sensíveis ao Contexto

A computação ubíqua, é uma visão que propõe novos paradigmas para a interacção com computadores e novas abordagens para a concretização desses paradigmas. Igualmente, assiste-se ao aparecimento de novas plataformas tecnológicas, como dispositivos móveis (*PDA's*, *Smartphones* e outros dispositivos com plataformas programáveis), dispositivos vestíveis e tecidos inteligentes, que começam a tornar possível a implementação dos desafios levantados pela computação ubíqua.

Uma das áreas onde se está a aplicar estes conceitos, é a informática médica, através da criação de ambientes inteligentes e de sistemas portáteis destinados à monitorização médica em ambiente controlado, como hospitais, ou como parte de um sistema de telemedicina.

A aplicação de conceitos da computação ubíqua em sistemas portáteis de telemedicina, poderia torná-los melhores em várias dimensões, mas quais conceitos introduzidos pela computação ubíqua poderão ser vantajosos neste tipo de sistemas? E de que forma? Como seria uma plataforma tecnológica que desse corpo a estes conceitos?

A computação ubíqua é uma área bastante vasta, como tal, o subgrupo de problemas da computação ubíqua a abordar nesta dissertação é delimitado na secção 3.1. Seguidamente, é justificado o potencial interesse da criação de um sistema computacional com estas características, comparando-o às soluções conhecidas existentes ao nível de investigação e da indústria (secção 3.2).

Seguidamente, é formulada a questão central desta dissertação (secção 3.3), assim como os desafios que levanta. Finalmente, são definidas as experiências e métodos que permitem avaliar os resultados e retirar conclusões relativamente às hipóteses levantadas neste trabalho.

### 3.1 Foco da Dissertação

Os novos dispositivos pessoais baseados em plataformas ubíquas, devido às suas características de mobilidade, usabilidade, conectividade e capacidade de processamento, têm potencial para permitir uma monitorização remota, inobstrusiva, permanente e em tempo real dos pacientes, onde quer que estes estejam.

Potencialmente, uma monitorização remota e eficiente, pode permitir que os utilizadores se dirijam menos vezes ao consultório médico, melhorando o aproveitamento de recursos para cuidados médicos e evitando o efeito de desertificação de vilas e aldeias carentes de assistência médica. Isto, tem efeitos sociais e económicos importantes, dadas as actuais tendências demográficas de envelhecimento da população, verificadas por exemplo, na Europa.

Estes sistemas, abrem novas possibilidades nos sistemas pessoais de telemedicina, no entanto, a computação ubíqua pode ser explorada de modo a melhorar estes sistemas.

Um dos conceitos introduzidos pela computação ubíqua, é a sensibilidade ao contexto, que explora sobretudo, problemas relacionados com a gestão de uma grande quantidade de informação em tempo real proveniente do ambiente obtida através de diferentes fontes de informação. Um dos problemas derivados, são os contextos multi-sensor, cenários nos quais é necessária a inferência de dados de múltiplas fontes de informação.

Alguns dos projectos de investigação e literatura associada a este tópico, fornecem um conjunto de ferramentas e abordagens que podem ser utilizadas para este fim, utilizando sobretudo, técnicas estatísticas ou técnicas baseadas em conhecimento.

Estes conceitos, poderiam ser utilizados para a realização do diagnóstico médico em tempo real, visto que este é um caso específico de um problema de inferenciação de múltiplas fontes de informação. O diagnóstico médico é, por sua vez, um problema amplamente estudado na medicina electrónica. Tal como nos contextos multi-sensor, podem ser utilizadas para resolver este problema, técnicas de caixa branca, cujos processos de decisão são visíveis e alteráveis e técnicas de caixa negra, cujos processos não são visíveis nem alteráveis. As técnicas de caixa negra são tipicamente as estatísticas.

Pretende-se nesta dissertação, a utilização das técnicas de caixa branca, visto que abrem outras possibilidades relativamente à manipulação dos processos de decisão, levantando a hipótese destes poderem ser alterados dinamicamente.

Nesta dissertação, o foco é então, a criação de um sistema de inferência de diagnóstico de caixa branca num sistema de computação ubíqua de telemedicina e a exploração dos benefícios que esta abordagem traz.

## 3.2 Trabalho relacionado

Existem sistemas que não utilizam os princípios ubíquos, que são bastante utilizados em telemedicina. Os novos sistemas que começam a utilizar princípios ubíquos, tornam os sistemas de monitorização mais usáveis, móveis e como tal, abrem possibilidades para a monitorização médica inobstrusiva. A sensibilidade ao contexto não é muito explorada em sistemas de telemedicina, mas é explorada noutros sistemas pessoais de saúde. Existe então, uma lacuna que pode ser preenchida com os princípios propostos ao longo deste trabalho.

Nesta secção, pretende-se apresentar outros sistemas espalhados pelas várias áreas que este sistema abrange, de modo a justificar os objectivos deste trabalho.

### 3.2.1 Sistemas Pessoais de Telemedicina

Os sistemas de saúde pessoais com funcionalidades de telemedicina que foram analisados na secção 2.3.2 e que não dispõem de características da ubiquidade, são bastante restringidos. O principal problema da não ubiquidade, deriva do facto do sistema depender da interacção do utilizador para obter os dados médicos, ou seja, estes dispositivos não são autónomos o suficiente, para obter dados médicos sempre que necessário. Isto, tira-lhes a aplicabilidade à detecção de situações de emergência, visto que se um utilizador tiver algum problema incapacitante, como perda de consciência, não poderá activar o dispositivo, activar canais áudio de emergência ou pedir qualquer tipo de ajuda. Outro dos problemas encontrados nos dispositivos, é a falta de mobilidade e o facto de obrigarem os utilizadores do sistema a estarem próximos do dispositivo para que ele possa ser utilizado.

### 3.2.2 Sistemas Ubíquos de Telemedicina

Existem várias abordagens em sistemas de monitorização portáteis ubíquos, baseadas em diferentes plataformas, como dispositivo móveis e dispositivos vestíveis. As abordagens com dispositivos móveis, nos quais estes são utilizados como *gateway* dos dados de sensores, têm algumas lacunas (secção 2.3.3.1 e secção 2.3.3.2).

Estas abordagens, implicam o envio constante de dados do utilizador, o que leva a que dados pessoais não significativos, sejam enviados, expondo a privacidade do utilizador mais do que o necessário. O envio constante dos dados, também levanta problemas de origem técnica, nomeadamente, problemas energéticos e provavelmente financeiros, devido à quantidade de dados transmitida pelas redes celulares. O custo energético do sistema também aumenta, o que pode trazer problemas de usabilidade, visto que resultados disponíveis do projecto *HealthService24*, demonstram que a utilização do sistema não foi possível durante mais de três horas, devido ao facto de que os dispositivos móveis utilizados neste projecto, tinham uma autonomia limitada. Por outro lado, a ausência de

qualquer capacidade de análise dos dados no sistema móvel, implica que esta esteja toda do lado ou dos profissionais de saúde, ou noutros nós de processamento da rede, o que levanta problemas de escalabilidade.

Na secção 2.3.4, são também dados a conhecer alguns sistemas que possuem capacidades de análise locais nos dispositivos. Estes sistemas não são, no entanto, adaptáveis, possuindo níveis de alarme predefinidos e regras que permitem avaliar dados de diferentes fontes de dados, também elas estáticas e predefinidas.

### 3.2.3 Sistemas Ubíquos Médicos Sensíveis ao Contexto

Na secção 2.3, verificou-se que existem sistemas que utilizam princípios ubíquos, mas que não se enquadram num sistema de telemedicina, porque não permitem qualquer tipo de interacção remota. Na sua generalidade, estes sistemas são bastante inovadores em termos de usabilidade, baseiam-se em dispositivos vestíveis ou em tecidos inteligentes, mas também por essa razão, não foram pensados para ter nenhum tipo de interacção remota, não sendo portanto sistemas de telemedicina. Estes sistemas, são o sistema desenvolvido no projecto *MYHEART* e o sistema *AUBADE*. Estes exemplos são muito inovadores, em termos de detecção de contextos, no entanto, não fornecem um sistema integrado de monitorização que possa ajudar realmente os utilizadores em caso de necessidades, reduzindo a sua utilização a monitorizações médicas em ambientes controlados, ou a fins de prevenção, com intervenção de âmbito local, sem qualquer tipo de supervisão médica.

## 3.3 Tese

Os processos de decisão de um sistema de um sistema de saúde pessoal ubíquo, com capacidade de diagnóstico médico e adaptável, podem ser alterados e evoluídos de forma remota.

A hipótese de tese acima referida, afirma várias premissas. Estas premissas não têm um significado muito preciso e como tal, é necessário explicitar o significado de cada uma delas. Ao longo das próximas secções, ir-se-á definir cada uma das premissas e o que cada uma delas quer dizer exactamente no âmbito deste trabalho.

### 3.3.1 Sistema de Saúde Pessoal Ubíquo

Dispositivo ubíquo, refere-se a um sistema *sempre presente e invisível* ao utilizador de um ponto de vista de usabilidade. O sistema de saúde pessoal, consiste no caso deste trabalho, num dispositivo móvel, que monitoriza o utilizador recorrendo a um conjunto de sensores médicos. Dada a grande quantidade de dispositivos móveis acessíveis no

mercado, e das suas características de conectividade e capacidade de processamento este dispositivo, será baseado num *Smartphone*, ou *PDA*.

### **3.3.2 Sensível ao Contexto e com capacidade de Diagnóstico Médico**

A sensibilidade ao contexto médico, refere-se à recolha de dados médicos dos sensores, análise desses dados e a inferência de contextos de alto nível e reacção aos resultados dessa inferência, em tempo real. O diagnóstico médico, é um desses contextos de alto nível e resulta na classificação do estado de prioridade do estado de saúde do utilizador.

### **3.3.3 Adaptabilidade dos Processos de Decisão**

A adaptabilidade dos processos de decisão, refere-se à possibilidade do sistema se poder adaptar às particularidades médicas de cada utilizador, por exemplo, ser capaz de se acrescentar a monitorização de certas patologias específicas de um utilizador, poder calibrar os valores limite de sinais vitais para cada um dos utilizadores e poder definir o processo de análise e recolha dos dados, para cada patologia. A adaptabilidade, irá ser conseguida através da utilização de uma técnica baseada em conhecimento para a realização do diagnóstico médico. Este facto permite que os processos de decisão possam ser “observados”, compreendidos e alterados.

## **3.4 Resultados Esperados e Métodos de Validação**

Este trabalho, ir-se-á concretizar na implementação de um protótipo que valide as ideias propostas na secção 3.3.

Este protótipo, deverá ser constituído por um dispositivo móvel que se liga a vários sensores e que tem capacidades de diagnóstico sobre os dados desses sensores. O diagnóstico, deve funcionar em tempo real, à medida que são obtidos os dados dos sensores. Este protótipo, deve ter capacidade de se ligar a uma segunda entidade através da *Web*, a partir da qual, adquire o conhecimento médico integrando-o, e provando a tese levantada neste trabalho. Os problemas esperados, são na interpretação do conhecimento médico, na integração do conhecimento e na integração dos restantes recursos, tais como sensores.

A validação desta tese, idealmente, seria realizada através da experimentação intensiva em ambiente real deste protótipo. Apesar de não ser possível uma experiência destas, realizou-se uma experiência que se repartiu entre ambiente controlado e ambiente real no âmbito do projecto, que deu origem a este trabalho, com um número pequeno de utilizadores e de profissionais de saúde. A recolha de dados, baseou-se em depoimentos conseguidos através de questionários, de modo a obter avaliações qualitativas por parte dos utilizadores e profissionais de saúde, relativamente às ideias por ele propostas. Esta experiência está descrita na secção 5.2.

## Diagnóstico em Sistemas Sensíveis ao Contexto

Para além das experiências, existirão ainda os testes em ambiente de desenvolvimento, que permitirão recolher alguns dados relativamente à validação desta tese e ainda, pequenos casos de estudo para a suportar.

## Capítulo 4

# Sistema Pessoal de Saúde com Capacidades de Diagnóstico

A solução proposta, constitui um sistema de telemedicina que tem como base um *Smartphone*, que interage com um conjunto de sensores no corpo do utilizador e analisa os dados dos sensores de modo a obter o diagnóstico em *tempo real* e a adaptar o seu comportamento, ao estado de prioridade do estado de saúde do indivíduo. Isto é visível na figura 4.1. O dispositivo interage com um servidor, que deve estar sempre acessível, de onde obtém a informação de contexto específica de cada utilizador. Esta informação contém, os valores limite, processos de diagnóstico, agenda de monitorização do indivíduo e os sensores específicos do indivíduo. O facto de se poder utilizar informação específica para cada utilizador, podendo-se adaptar um comportamento diferente para cada um destes, torna o sistema *adaptável*. O dispositivo interage com o servidor, também para enviar a informação sobre o estado de saúde do indivíduo.

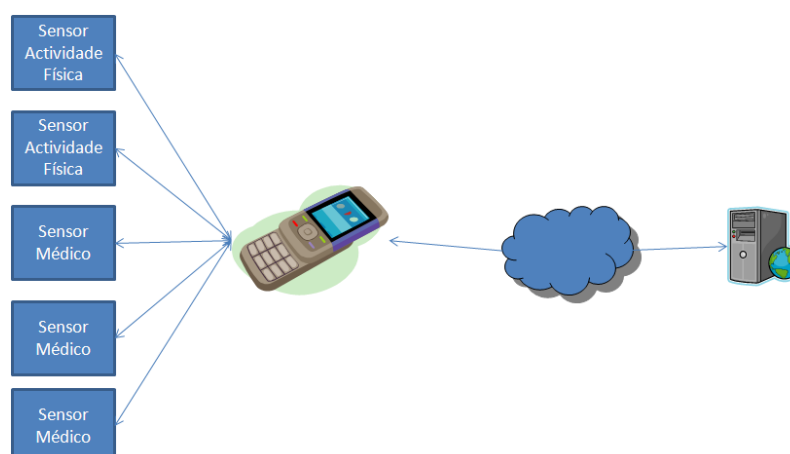


Figura 4.1: Visão Geral da Solução

## 4.1 Especificação do Sistema Ubíquo

### 4.1.1 Visão Geral

As funcionalidades do sistema, encontram-se divididas em quatro pacotes lógicos: Interação com o dispositivo, sensibilidade ao contexto, administração da rede de sensores e gestão da informação médica. O pacote da interação, contém todas as funcionalidades respeitantes à interação e usabilidade do sistema e o pacote de sensibilidade ao contexto contém as funcionalidades respeitantes à aquisição e inferência de contexto, a partir dos dados das diferentes fontes de informação. Contém também, as funcionalidades que definem o comportamento da aplicação em relação ao contexto. Existe ainda o pacote de administração da rede de sensores, que contém funcionalidades para gerir os sensores da *BAN* do utilizador. Existem dois actores no sistema: o utilizador alvo da monitorização médica, e os processos automáticos do sistema. Isto é visível na figura 4.2

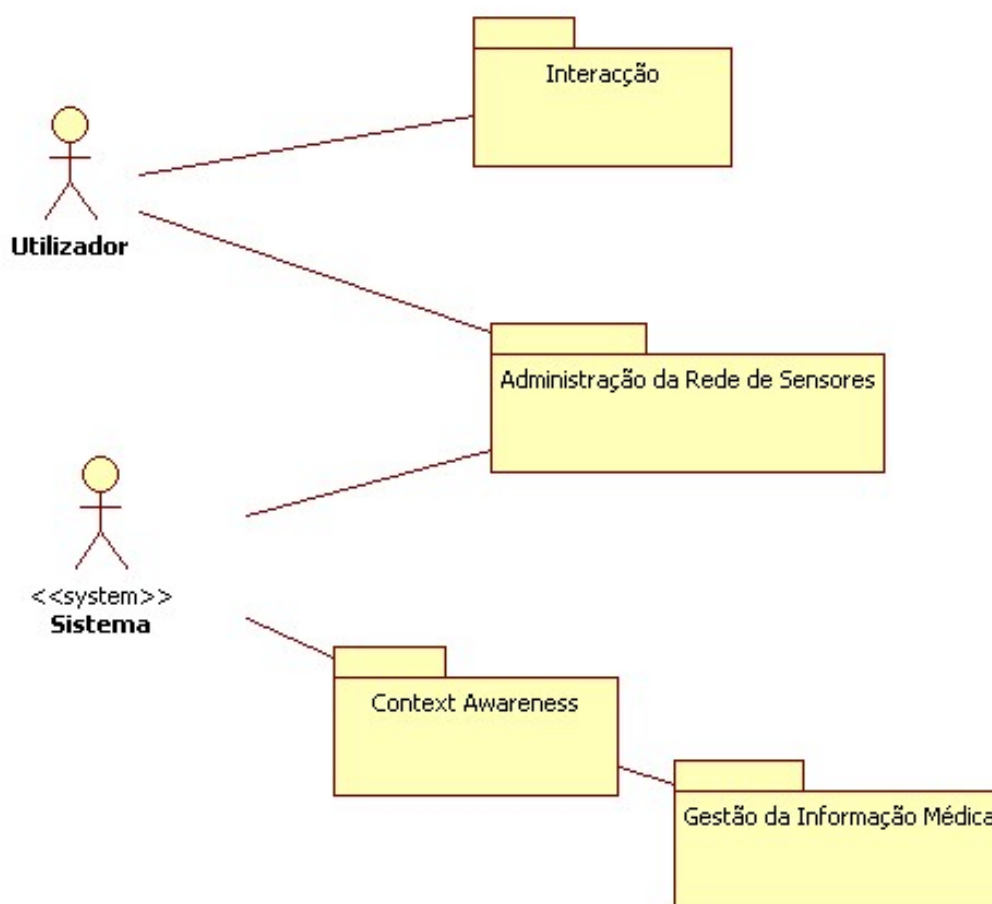


Figura 4.2: Visão Geral dos Casos de Uso

#### 4.1.2 Interação

A interação, por definição da computação ubíqua, pretende-se que seja o mais simples e transparente possível, de modo a tornar o sistema o mais inobstrusivo possível. Neste sistema, a interação necessária com o *Smartphone* reduz-se ao seu transporte físico, porque se pretende que o sistema seja completamente autónomo. Por razões de aplicabilidade, é necessária a existência de um *panic-button* no sistema, que permita avisar os profissionais de saúde de que o utilizador necessita de ajuda, a pedido deste. Este é o único caso de uso neste pacote de funcionalidades, tal como é visível na figura 4.3.

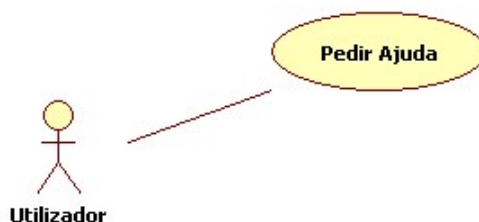


Figura 4.3: Casos de Uso do pacote de Interação

#### 4.1.3 Sensibilidade ao contexto

A sensibilidade ao contexto significa que o sistema, de alguma forma, utiliza o contexto na sua operação. Os contextos que este sistema irá utilizar são o contexto médico, contextos de actividades físicas e contextos geográficos. A informação de base para a inferência do contexto médico são os dados dos sensores médicos, e esta inferência é realizada através de um diagnóstico médico sobre esses dados. Os diagnósticos, podem ainda ser influenciados pela actividade física do utilizador, como tal, o sistema contém funcionalidades para integrar informação destes contextos. O diagnóstico médico, é realizado com base no conhecimento médico específico do utilizador e resulta na atribuição de uma classificação ao estado de saúde do paciente, que por sua vez, pode resultar no envio de um alerta, que pode agregar uma observação. O contexto geográfico, consiste na avaliação do posicionamento do utilizador, de modo a verificar se este se encontra longe de um conjunto de posições, tidas como significativas na vida diária do utilizador. Se o utilizador se encontrar longe destas posições, o seu posicionamento passa a ser enviado para o servidor de informação médica e prologa-se até que o utilizador volte a estar ao alcance dos pontos significativos. Isto é visível na figura 4.4.

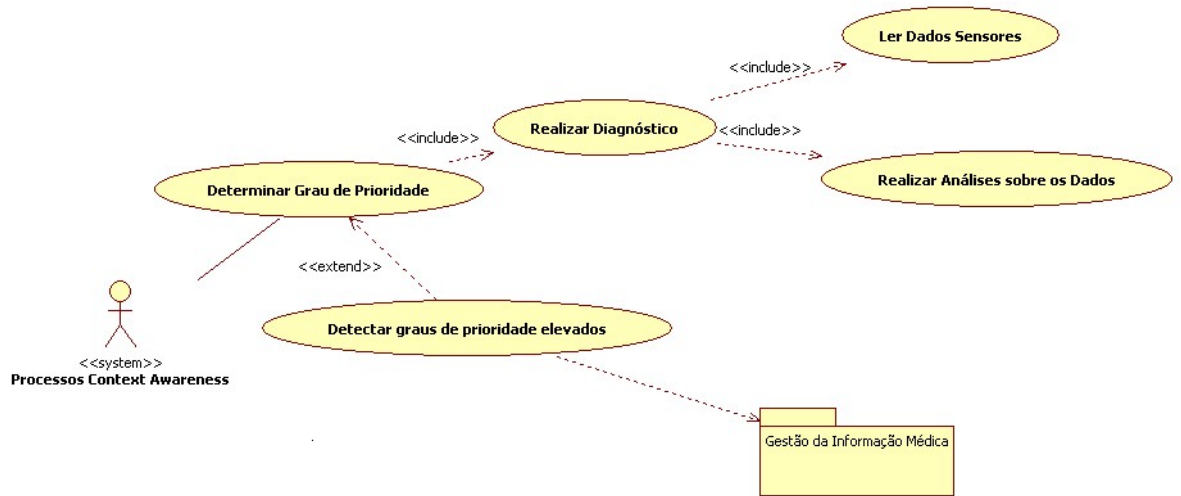


Figura 4.4: Casos de Uso da Sensibilidade ao Contexto

#### 4.1.4 Gestão da Rede de Sensores

A gestão da rede de sensores contém funcionalidades para a adição, remoção e descoberta dos sensores. Para que um sensor possa funcionar com a rede de sensores deste sistema, tem de ser “conhecido” pela rede, o que se concretiza na necessidade da existência de um conjunto informação do lado do sistema sobre o sensor, que agrega, a *identificação* do sensor, que serviços podem ser obtidos do sensor, que configurações utilizar e que *drivers* utilizar para operar com os sensores. Esta informação, que descreve os sensores do utilizador, pode ser recebida através do servidor de contexto, ou pode ser especificada pelo próprio utilizador do sistema. Esta informação é fornecida no registo do sensor no sistema. A descoberta dos sensores, serve para associar esta informação a um dispositivo real que esteja ao alcance do dispositivo móvel. Estes casos de uso, são visíveis na figura 4.5

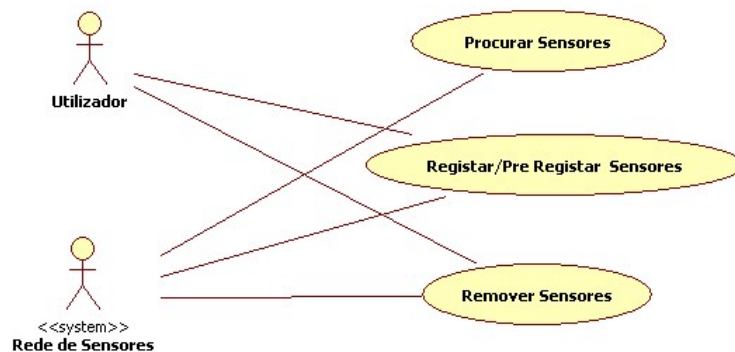


Figura 4.5: Casos de Uso do Pacote de Gestão de Sensores

#### 4.1.5 Gestão da Informação do Utilizador

A gestão de informação do utilizador, consiste fundamentalmente, na sincronização da informação médica do utilizador, com o servidor de contexto. Esta sincronização, abarca a recepção das posições significativas do utilizador, da descrição dos sensores do utilizador e da informação médica do utilizador. Esta informação médica consiste nos processos referentes ao diagnóstico do utilizador. Este pacote, abarca também, as funcionalidades de envio de alertas e observações médicas. Os alertas, correspondem a todas as situações em que seja detectada uma situação médica que necessita de atenção e podem transportar observações, ou não. As observações, constituem um conjunto de dados médicos e análises médicas recolhidas, ao qual se adiciona uma classificação do estado de saúde do utilizador e, se disponível, a posição georeferenciada onde a observação foi recolhida. As observações, estão associadas a uma determinada patologia e os dados médicos que fazem parte da observação, são os considerados significativos para essa patologia que se pretende observar. O objectivo, é que os médicos possam tomar as decisões o mais rápido possível, como tal, a informação disponível numa observação, deve ser apenas a significativa. Para além de poderem ser enviadas com alertas, as observações podem ser enviadas como rotina, de acordo com uma agenda definida para o utilizador. Estes casos de uso, encontram-se demonstrados na figura 4.6.

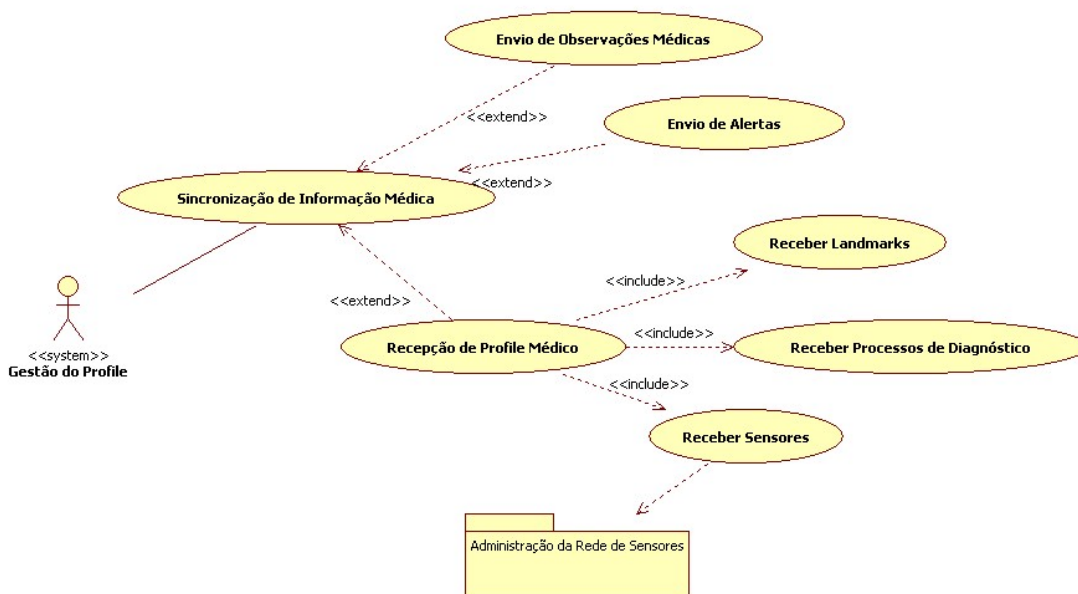


Figura 4.6: Casos de Uso do Pacote de Gestão da Informação Médica

## 4.2 Sensibilidade ao Contexto

### 4.2.1 Contextos Utilizados

A sensibilidade ao contexto nesta aplicação, é estruturada por um conjunto de contextos com diferentes complexidades, que podem ser descritos por uma pilha, sendo o nível mais baixo o dos dados que resultam dos sensores. O nível mais alto desta pilha, é o contexto de diagnóstico, que depende de relações complexas entre os diferentes sensores, a informação de contexto específica do utilizador e os contextos de actividade física. Estes contextos de actividade física, não têm uma influência tão directa no processo de decisão relativo ao diagnóstico, no entanto influenciam-na indirectamente, porque por exemplo alteram os valores limite dos sinais vitais. A pilha de contextos é visível na figura 4.7.

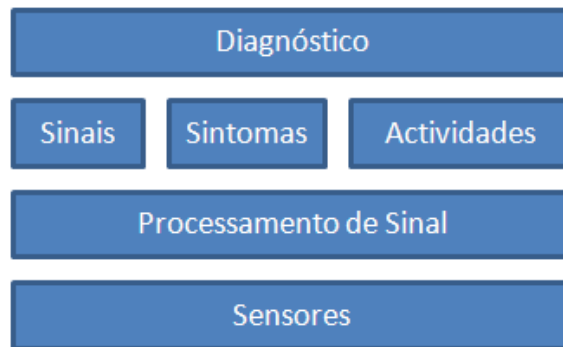


Figura 4.7: Pilha de Contextos Utilizada

### 4.2.2 Identificação do Contexto de Diagnóstico

O diagnóstico, idealmente, deve ser realizado em tempo real a partir dos dados dos sensores, o que significa que este deve ser realizado relacionando vários fluxos contínuos de informação.

O problema da extracção de padrões complexos de fluxos contínuos de informação, é um problema abordado na computação ubíqua, aquando, por exemplo, na inferênciação num contexto multi-sensor.

Na computação ubíqua, existem várias abordagens para resolver este problema, as quais, se podem dividir em dois grandes grupos: abordagens estatísticas e abordagens baseadas em conhecimento, como ontologias. A aquisição e análise de dados contínua consome muitos recursos, porque obriga a que sensores e transmissão de dados funcionem constantemente, assim como tecnicamente, a integração da análise contínua de dados integrando conhecimento, é um problema difícil de resolver.

Uma das soluções mais utilizadas, é realizar o diagnóstico apenas sobre os eventos conhecidos, ou sobre as mudanças do contexto. No caso deste sistema, esta não é no entanto uma solução completa, porque verificou-se que nem todos os sensores suportam um

funcionamento *event-driven*, ou seja, nem todos são suficientemente autônomos para iniciar a comunicação apenas quando algo significativo acontece, e para estes, é necessário utilizar um paradigma *pooled*. Se estes sensores, não forem interrogados vezes suficientes, pode acontecer a perda de algo significativo sobre o estado de saúde do utilizador. A definição da frequência de *pooling*, no entanto, é um problema parcialmente resolvido com eficiência no domínio médico, visto que na maior parte dos sinais vitais, existe uma frequência mínima para a obtenção de medidas de sinais vitais e os profissionais de saúde têm a opinião, que esta frequência é suficiente. Apesar disto, existem sinais vitais que devem ser monitorizados permanentemente. Não é viável, deixar os médicos fora do processo de monitorização, sendo importante que pelo menos, seja enviada informação de rotina periodicamente.

A abordagem proposta, consiste em que a recolha dos dados seja condicional e completamente determinada pelo conhecimento da aplicação. Com base no conhecimento disponível no sistema, é utilizado um raciocínio dedutivo que segue uma árvore de decisão e que determina que dados devem ser recolhidos, em que ordem e que tipo de análises devem ser utilizadas para obter os resultados. Através deste raciocínio, pode também ser decretada a necessidade de que os dados de um sensor sejam analisados por um período de tempo, ou mesmo, permanentemente. Nesta ordem, apenas são recolhidos os dados considerados significativos, poupando assim nos recursos necessários para realizar a análise de diagnóstico. Este raciocínio dedutivo, começa com um evento derivado dos sensores, de eventos despoletados por agendas especificadas para verificar uma determinada patologia, ou ainda, por um evento despoletado por uma análise contínua.

Deste modo, com base em algum conhecimento médico, foram consideradas três níveis possíveis para as frequências de verificação das diversas patologias: periódicas, permanentes e inexistentes. As frequências permanentes, aplicam-se a sinais médicos que necessitem de ser avaliados permanentemente, as frequências inexistentes, aplicam-se a sinais cujos sensores sejam completamente autônomos. As frequências periódicas, aplicam-se quando existe uma frequência recomendada para a verificação de um sinal médico, o que acontece muito frequentemente. Esta informação, irá ser embebida no conhecimento médico que determina a forma de recolha dos dados e deste modo, obtém-se uma solução completa e configurável para os requisitos especificados. Um exemplo de funcionamento deste mecanismo é visível na figura 4.8.

Outro dos desafios desta abordagem, é a integração de informação temporal, visto que muitos dos diagnósticos médicos, têm uma dimensão temporal. Neste cenário, são possíveis casos em que certas condições do diagnóstico só se verificam passado algum tempo e casos em que certas recolhas têm de ser repetidas passado alguns instantes. Existem ainda casos, em que certas combinações de condições têm de estar separadas por intervalos de tempo, para representarem algo significativo.

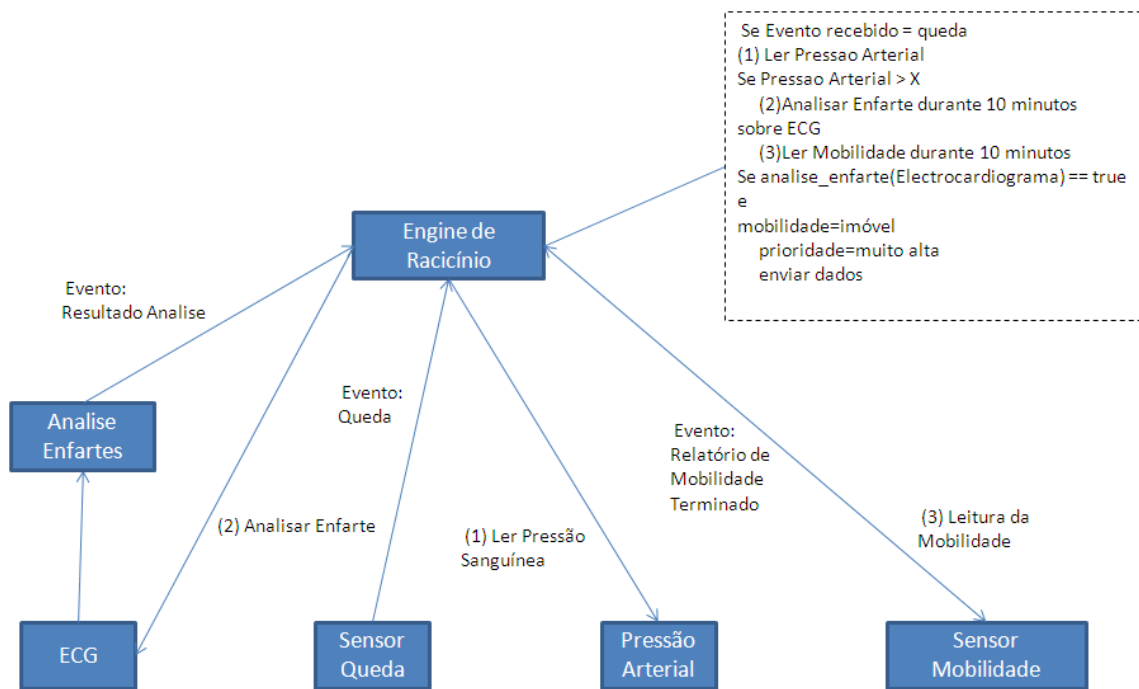


Figura 4.8: Exemplo do funcionamento do Motor de Raciocínio

### 4.3 Arquitectura

Esta secção, destina-se a demonstrar qual a estrutura do sistema de *software*, na qual irão ser aplicadas as ideias citadas nas secções anteriores.

#### 4.3.1 Requisitos

**Tolerância à falha e Eficiência Energética** Cada um dos componentes está exposto a falhas devido à interação com outros dispositivos externos. É importante garantir que um componente, não prejudica o funcionamento de outros.

#### 4.3.2 Arquitecturas semelhantes

Uma das soluções arquitectónicas para sistemas de computação ubíqua, é desenvolver uma que resolva o problema específico, perdendo-se no entanto, a adaptabilidade e a capacidade de evolução. Algumas das outras abordagens mais utilizadas, dividem-se em Arquitecturas de agentes e na utilização de arquitecturas orientadas aos serviços, com um *middleware* que garanta a comunicação entre os vários componentes e serviços.

##### 4.3.2.1 Arquitecturas de Agentes

As arquitecturas orientadas aos agentes, podem ser aplicáveis em arquitecturas sensíveis ao contexto, devido ao facto de que conceptualmente, cada recurso do sistema pode ser

representado através de um agente, que interage com outros de modo a identificar o contexto através da passagem de mensagens, com completa indirecção nas interacções realizadas entre todos os agentes. Existem algumas plataformas que podem ser utilizadas para este fim segundo [Dey00], nomeadamente, *Open Agent Architecture* [CM01], *Hive* [MGR<sup>+</sup>99], *MetaGlue* [Phi99].

#### 4.3.2.2 Arquitecturas de Componentes

As arquitecturas orientadas aos serviços, são arquitecturas que são compostas por componentes que são acedidos por outros, dispostos numa estrutura mais ou menos complexa. Os serviços no âmbito de uma arquitectura sensível ao contexto, podem estar disponíveis remotamente ou localmente e podem ser sensores, ou análises que dependam de vários sensores. A integração entre os vários serviços, pretende-se transparente e é conseguida através de um componente chamado *middleware*. Existem variados tipos de *middleware* e variadas implementações utilizando diferentes abordagens de associação e comunicação entre os vários componentes da aplicação.

#### 4.3.2.3 Middleware

O *middleware*, é um componente que se destina à integração de componentes garantindo-se a comunicação e integração entre os mesmos de modo muito simplificado e transparente, sendo portanto, muito utilizado em sistemas distribuídos, o que inclui os sistemas ubíquos. Para dispositivos móveis, existem vários tipos de *middleware*, os quais suportam diferentes tipos de funcionalidades relacionadas com o modo de integração entre os componentes [Kja07].

- *Middleware* reflectivo
- Repositório de Tuplos partilhados
- *Middleware* sensível ao contexto
- *Middleware* Orientado aos eventos

***Middleware* reflectivo** A reflexão, é um tipo de mecanismo que permite a um componente de software a introspecção da sua estrutura e a alteração do comportamento, em tempo de execução. Este mecanismo, está presente em linguagens de programação, sistemas operativos e mais recentemente, em sistemas distribuídos.

Um *middleware* deste género, é sempre composto por dois níveis um nível de meta-informação, e um nível de base. O nível de meta-informação define os comportamentos dos componentes do nível de base durante a execução, podendo este ser alterado durante a execução.

Isto permite que o *middleware* se possa adaptar e reconfigurar em relação às mudanças do ambiente, otimizando a sua estrutura.

**Tuplos partilhados** O sistema de tuplos partilhados consiste numa memória partilhada entre os vários sistemas distribuídos. Cada um dos componentes do sistema, acede ao repositório alterando os dados lá presentes. Nesta situação, o repositório é o único componente conhecido pelos outros componentes, para além deles próprios.

**Middleware sensível ao contexto** O *middleware* sensível ao contexto, é capaz de integrar transparentemente informação de contexto no sistema. O sistema, seguidamente adapta o seu funcionamento à informação de contexto disponível.

**Middleware orientado aos eventos** O *middleware* orientado aos eventos, consiste no princípio fundamental da comunicação assíncrona através de mensagens entre os vários componentes do sistema. Os componentes são associados entre si através do paradigma *publish/subscriber*. Este tipo de *middleware*, permite que os componentes sejam completamente independentes tanto em termos de estrutura, como em termos de funcionamento em tempo de execução.

### 4.3.3 Integração de Sensores através da Web

A *OGC Sensor Web Enablement*, é uma *framework* de melhores práticas, proposta pela *OGC*[OGC] tendo em vista o controlo e operação de sensores dispostos pela *Web* e a partilha de informação entre os mesmos. A *OGC Observations and Measurements*, é um dos *standards* propostos pela *OGC*. Neste *standard*, é proposto um modelo de dados e melhores práticas de desenho para a representação de dados em sistemas baseados em sensores, assim como os *encodings*, utilizando a tecnologia, *XML* para esses dados. O *Sensor Modeling Language*[[ea](#)] é também um *standard* proposto pela *OGC* e foi utilizado no contexto desta arquitectura.

#### 4.3.3.1 Observações e Medidas

Este *standard* propõe duas entidades base: *Measurements* (Medidas), *Observations* (Observações). Este modelo de dados, é suficientemente flexível para ser adaptado ao domínio médico e para ser utilizado como referência para os componentes a serem utilizados no contexto desta aplicação.

**Observações** As Observações, são um conceito definido no *standard Observations and Measurements*, como um conjunto de dados que contém resultado associados a um determinado fenómeno. No contexto do *standard*, uma observação é um caso específico de um

evento, o que significa que está associado a um instante específico de tempo. Os eventos neste *standard*, têm também associado o seu geoposicionamento e a geometria ao qual se referem.

**Measurements** As *Measurements*, são descritas como um caso específico de Observação e correspondem a um conjunto de informação direcionado para codificar qualquer tipo de amostras dos sensores.

**Observation Procedures** Os *Observation Procedures*, são definidos como pertencentes às Observações e definem o método utilizado, para obter o resultado da Observação. Os *Procedures* podem ser complexos, como por exemplo, sequências de operações, ou de outras observações, que permitem obter o resultado esperado relativamente a um fenómeno. Apesar disto, os *Procedures* não são uma linguagem suficientemente completa para descrever algoritmos. São utilizados para descrever um conjunto de passos, que podem ser sequências, que determinam todo o encadeamento de processos, assim como *inputs* e *outputs* necessários em cada passo, para obter um resultado final. Os *Observation Procedures* são normalmente descritos utilizando *Sensor Modeling Language*

#### 4.3.3.2 Linguagem de Modelação dos Dados dos Sensores

*Sensor Modeling Language*[[ea](#)] é outro *standard* da OGC[OGC], que propõe modelos descritivos de meta-informação para a representação de sensores. Estes modelos de meta-informação, suportam a especificação de informação relativa à descoberta, operação, geolocalização de medidas realizadas pelos sensores e medidas de *performance* dos sensores, de modo a criar um meio universal de interoperabilidade com sensores na *Web*.

Estes modelos, são especificados através de *schemas XML* e permitem a especificação de informação sobre as propriedades físicas medidas pelos sensores, sobre os atributos de qualidade dessas propriedades físicas e sobre as características das respostas dadas pelos sensores.

Como este *standard*, dirige-se particularmente à representação de informação geográfica e como tal, também permite a definição, das características geométricas dos vários tipos de dados observáveis dos sensores. Esta representação, é feita recorrendo a outro *standard* OGC.

#### 4.3.4 Aplicação dos standards OGC

Os *standards* OGC, foram utilizados neste sistema como *framework* conceptual.

As vantagens da utilização deste *standard* são as seguintes:

- Arquitectura mais perceptível, como todas as arquitecturas baseadas em padrões.

- Modelos de dados conhecidos, o que facilita a partilha de informação e a interface com novos componentes.
- Especificação completa para a todo o tipo de informação necessária no sistema: *amostras de sensores, descrição de sensores, descrição de processos recolha de informação*.
- Adaptabilidade do modelo de dados para outros domínios fora do geográfico.
- Um modelo de dados aceite como *standard* e boa prática.

Este *standard*, influenciou fortemente a arquitectura, no entanto, não se poderia utilizar tudo na íntegra, porque complicaria muito a implementação e também pelo facto, de apesar de ser bastante geral e adaptável, não era utilizável na íntegra no domínio médico. Este *standard*, foi influenciado pelo padrão de arquitectura *Observation*, proposto por Joseph Yoder[YJ02]. A utilização deste *standard*, verificou-se mais na especificação do modelo de dados do repositório utilizado na aplicação, embora tenha influenciado fortemente toda a arquitectura e desenho da aplicação. Segundo [Pro06], é possível mapear os *standards OGC* para ontologias, o que garante que é possível integrar conhecimento especificado desta forma.

#### 4.3.4.1 Padrões de Observação

Os padrões de observação, foram mapeados dos *Observation Procedures* e são conjuntos sequenciais de passos que descrevem como deve ser realizada a recolha de dados, a sua análise e que verificação de condições. Um padrão de observação está associado à Observação de um *senal* médico. Os *Observation Procedures* permitem também a associação dos passos ao domínio temporal, o que também foi utilizado, dado o facto da dimensão temporal ser considerada necessária para a realização de diagnósticos. Cada passo neste sistema, foi mapeado aos diversos tipos de serviços, que irão estar disponíveis na aplicação.

Cada utilizador, terá os seus padrões de observação, no entanto, para facilitar a criação de padrões de Observação, existe o conceito de *Templates de Observação*. Os *Templates de Observação*, são padrões de observação previamente calibrados para a verificação de sinais relativos a diversas patologias, como por exemplo, a diabetes.

#### 4.3.4.2 Observações

As observações, correspondem directamente ao mesmo conceito do *standard* da OGC, e são portanto, tratados como eventos. Foi aproveitado parte do modelo de dados das observações, para representar a informação julgada necessária neste sistema. Na prática,

contém todas as amostras ou resultados de análises, assim como uma classificação relativa à gravidade do estado de saúde do utilizador. Contém ainda um relatório, que contém as condições verificadas, quais as conclusões retiradas e o seu geoposicionamento. Uma observação, pretende-se que transporte toda a informação médica necessária, para possibilitar a conclusão rápida do médico, sobre o estado de saúde do paciente.

#### 4.3.4.3 Limitações encontradas

O *standard* permite a associação de *Observations* e *Procedures* a contextos espacio-temporais. Por outras palavras, é possível dizer que uma *Observation* e o respectivo *Procedure* são válidos apenas quando se encontra dentro de uma dada janela espacio-temporal. Neste sistema, é também necessário associar as *Observations* e *Procedures* a outros contextos, como contextos de actividades físicas, o que não se encontrava no âmbito deste *standard*. Estas dimensões foram adicionadas no modelo de dados utilizado nesta arquitectura.

#### 4.3.5 Arquitectura Lógica

A arquitectura geral, assentou numa arquitectura de componentes/serviços. Estes componentes fornecem serviços e comunicam através de *middleware*, havendo direcção total entre todos os componentes. O estilo do *middleware* escolhido é simultaneamente, orientado a eventos e repositório de tuplos. Isto, porque foi utilizada uma arquitectura estilo *blackboard* [Cor91]. Esta arquitectura, caracteriza-se por utilizar um *middleware*, simultaneamente *event-driven* e repositório de tuplos. As alterações verificadas no repositório de tuplos, são reportadas através de eventos aos outros componentes do sistema. Aplicou-se o padrão *publish/subscriber*, de modo a que cada componente só receba os eventos em que está "interessado". Isto é visível na figura 4.9.

Cada um dos componentes é completamente independente, utilizam o repositório de dados para efectuar alterações, enviam e recebem eventos. Cada um dos componentes não conhece os outros componentes e não conhece as implicações das acções que realiza sobre os dados. Isto corresponde à utilização de uma estratégia de *componentes soltos* (*loosely coupled components*). Para além da tolerância a falhas, (visto que cada um dos componentes é completamente independente esta valência melhora), outra razão que levou à escolha deste tipo de arquitectura, foi a possibilidade de poupar recursos energéticos, visto que os componentes só precisam de ser activados quando algum evento acontece.

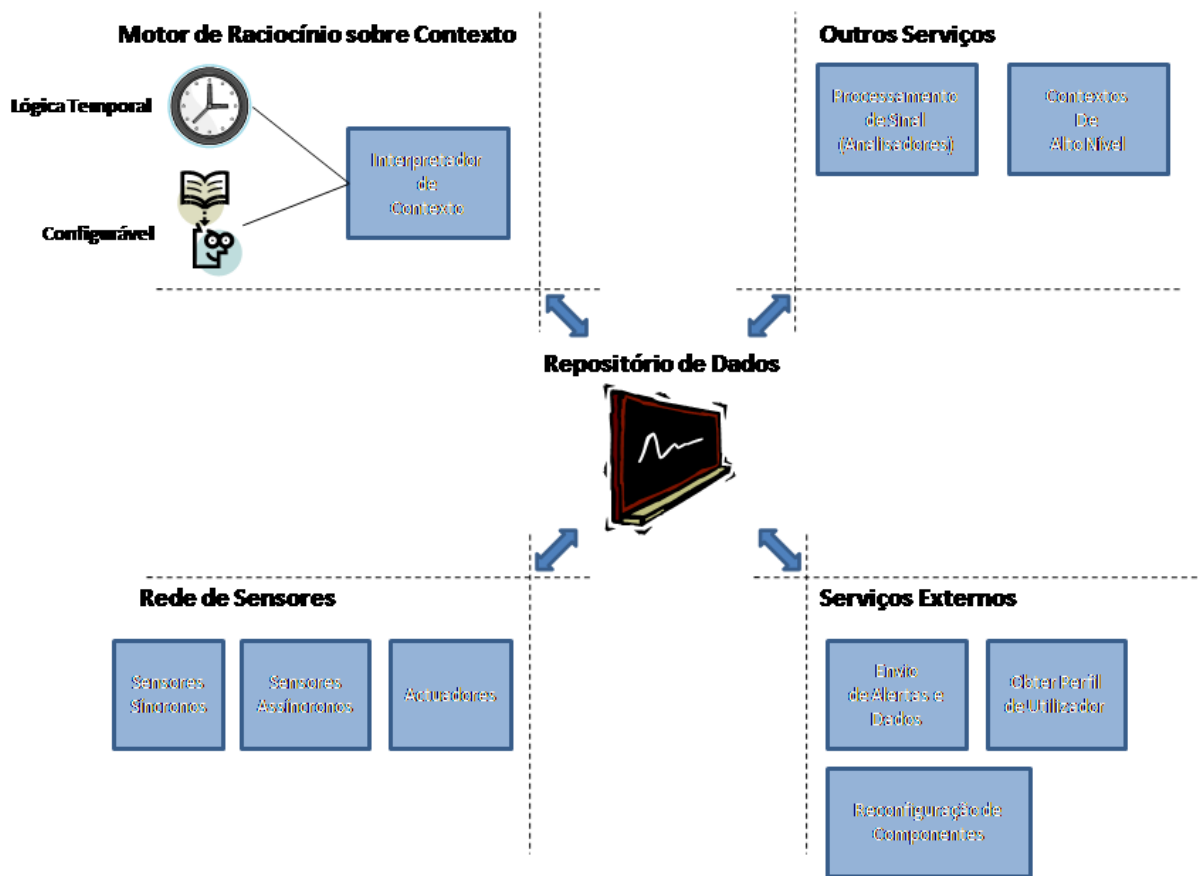


Figura 4.9: Arquitectura Geral - Estilo *Blackboard*

#### 4.3.5.1 Análises de Informação Médica

Os analisadores de contexto, são componentes que são capazes de extrair dados com significado a partir de dados médicos de mais baixo nível, para além do motor de raciocínio principal. São serviços disponíveis na aplicação, que são capazes de receber um conjunto de parâmetros e devolver resultados. Destinam-se a vários fins, tais como, analisar os resultados provenientes dos vários sensores, ou analisar um conjunto de amostras obtido no âmbito da execução de um padrão de observação.

#### 4.3.5.2 Rede de Sensores

A rede de sensores, é o componente que gere todos os sensores da aplicação. Dispõe de funcionalidades para descoberta, configuração e leitura dos sensores. Para tal, a rede

de sensores disponibiliza serviços, que internamente são mapeados para os sensores correctos e causam as necessárias acções de configuração e de activação dos sensores. Deste modo, para utilizar os sensores, basta aceder aos serviços fornecidos pela rede de sensores, para obter dados. Da mesma forma, implementa-se o padrão *Facade*[GHJV95].

A rede de sensores, gere sensores assíncronos, sensores que têm autonomia para iniciar a comunicação para reportar eventos e sensores síncronos, que necessitam que lhes sejam pedidos dados. Suporta também, sensores que sejam reconfiguráveis, ou seja sensores que transmitem dados continuamente e que podem sofrer alterações na sua configuração inicial.

A rede de sensores, é também capaz de procurar e registar sensores, que tenham sido pré-registados. O pré-registo, implica a especificação de um identificador que os permita descobrir e identificar no ambiente, assim como a especificação dos *drivers* que permitem operar e configurar os sensores e o conjunto de serviços que cada um deles oferece. A descoberta de sensores é feita periodicamente, com base na lista de sensores pré-registados. Este funcionamento é visível na figura 4.10.

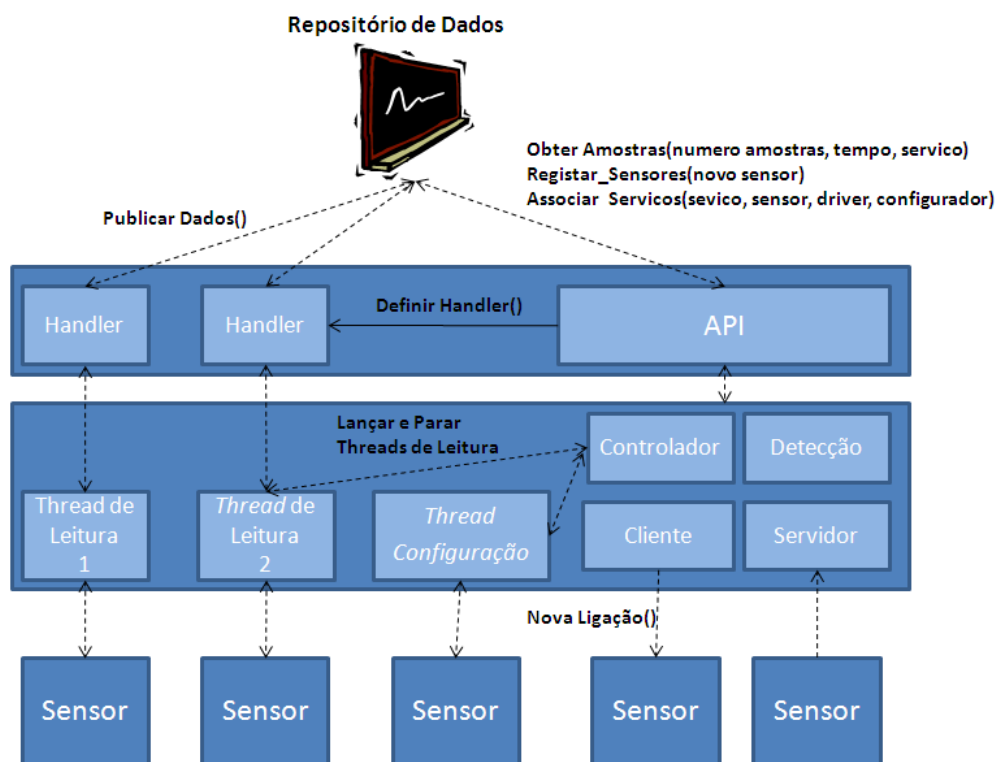


Figura 4.10: Funcionamento da Rede de Sensores

#### 4.3.5.3 Repositório

O repositório, é o componente responsável por todo o armazenamento de informação na aplicação, e é também capaz de reportar eventos sobre os fluxos de dados. O modelo

conceptual de dados presentes no repositório, é o modelo adaptável baseado no padrão *Observation* e no *standard OGC Observations and Measurements*. Isto é visível na figura 4.11.

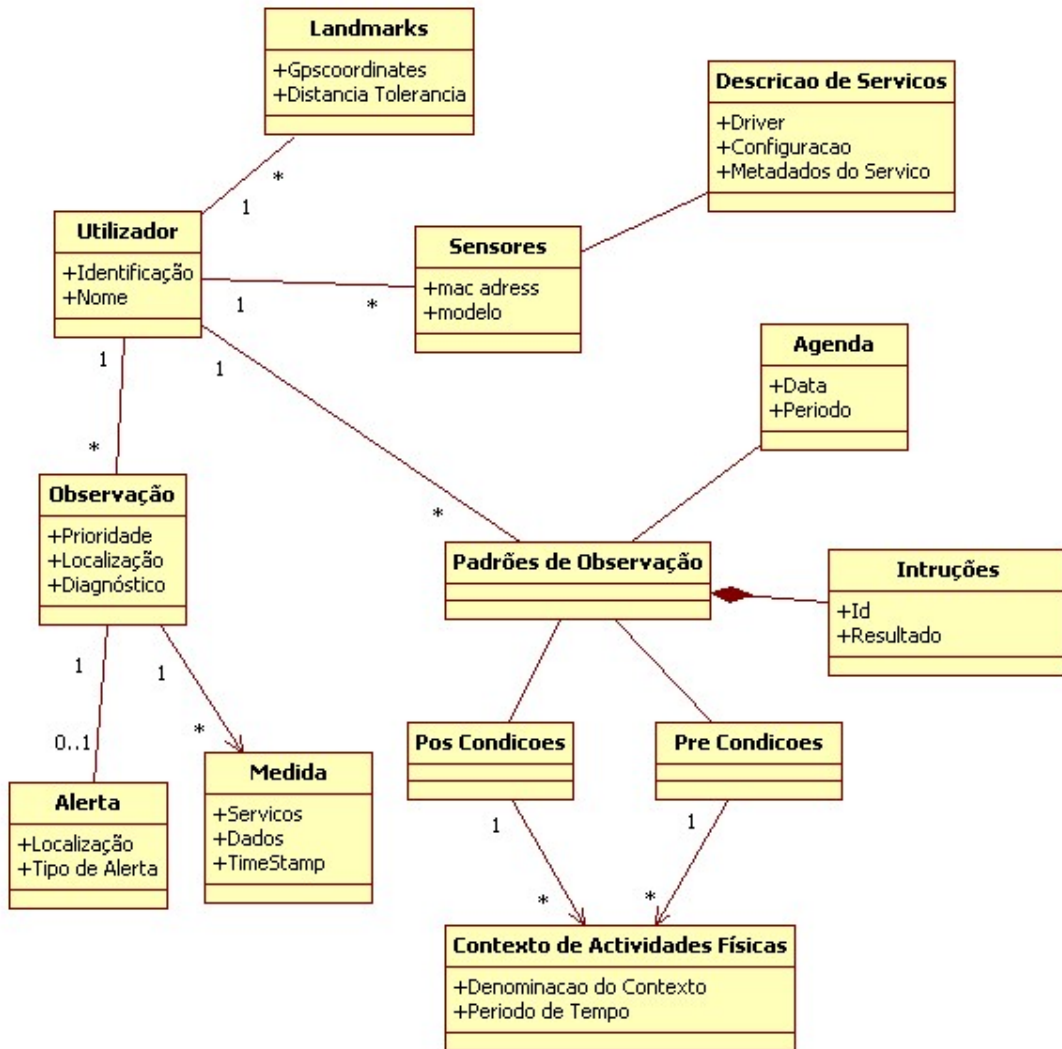


Figura 4.11: Modelo de Dados do Repositório

#### 4.3.5.4 Interpretador de Padrões de Observação

O interpretador de Padrões de Observação, é responsável pela interpretação da sequência de instruções dos mesmos. Os Padrões de Observação, são utilizados para definir o modo de realização dos diagnósticos, utilizando os sensores e analisadores disponíveis no sistema. Isto é visível na figura 4.12.

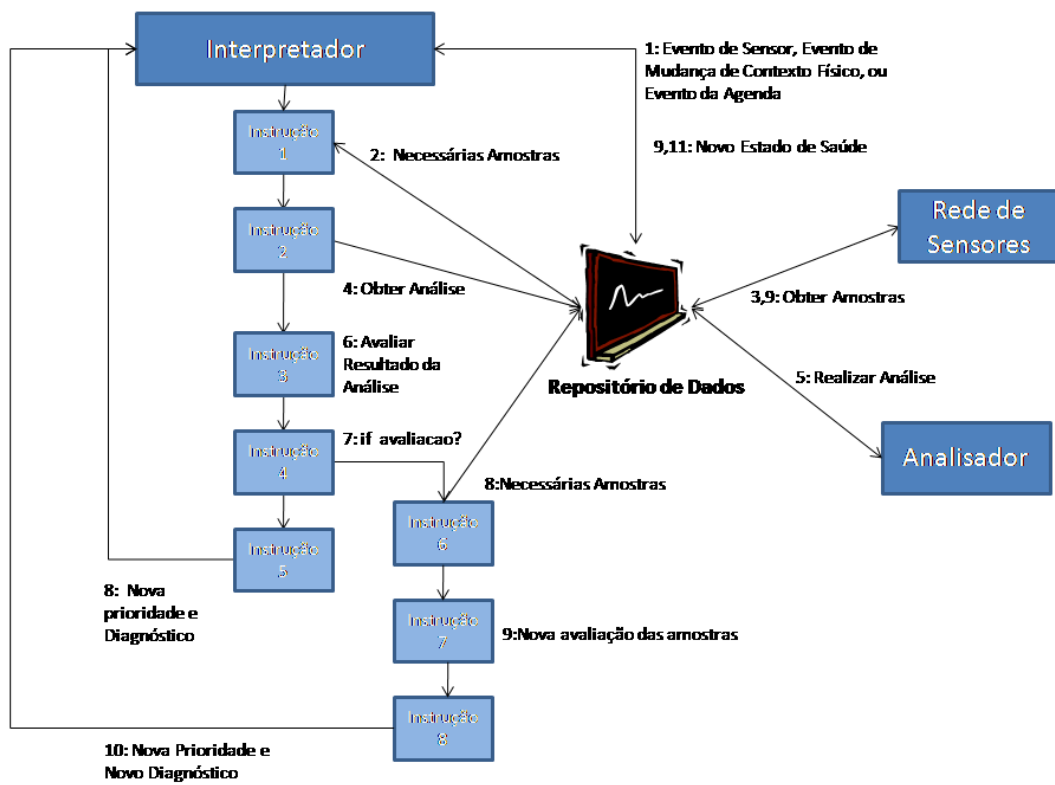


Figura 4.12: Funcionamento do Interpretador de Padrões de Observação

O contexto de execução, é visto como um conjunto de pré-condições que têm de se verificar, para que as regras possam começar a ser executadas. Estas pré-condições, consistem em contextos de actividade física e agendas. As agendas, determinam a frequência de execução e os instantes em que devem ser realizadas as observações.

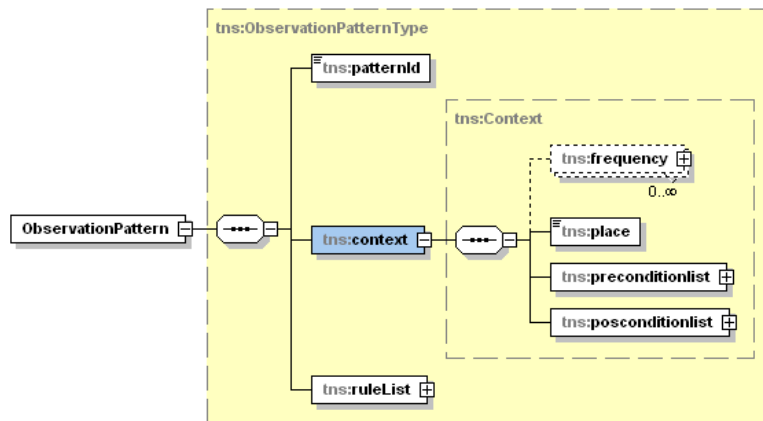


Figura 4.13: Modelo de Dados dos Padrões de Observação

A presença dos contextos de actividade física nas pré-condições, deve-se ao facto de que medicamente, os valores limite dos sinais vitais são afectados pela actividade física do utilizador. Isto é devido a que, por exemplo, para certas amostras do electrocardiograma o utilizador teve de estar em repouso durante dez minutos. Existem também situações, onde importa também que existam diferentes Padrões de Observação para diferentes actividades físicas. O modelo de dados dos padrões de observação e de toda a informação relacionada é visível na figura 4.13.

#### 4.3.5.5 Linguagem de Observação

A interpretação de Padrões de Observação, consiste na execução de uma sequência de instruções que resultam na obtenção de uma observação que representa o diagnóstico. Foi então criada uma linguagem, baseada no *standard SensorML* da *OGC*, para representar os processos de diagnóstico. Foi escolhido um *encoding* em *XML*, que permite representar todas as instruções. Este *encoding* é visível na secção 4.14.

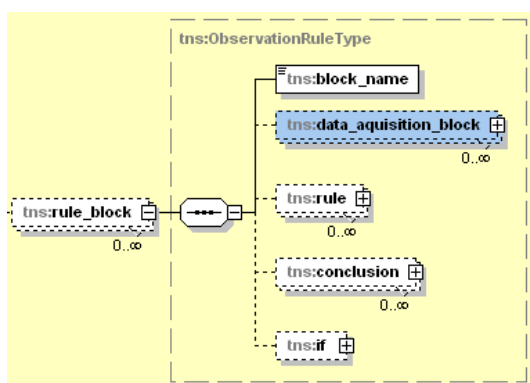


Figura 4.14: Visão Geral das instruções da linguagem dos Padrões de Observação

**Recolha de Amostras** Estas instruções permitem recolher um conjunto de amostras a partir dos sensores. O tamanho do conjunto pode ser delimitado pelo tempo, ou pelo número de amostras pretendidas. Estas instruções contêm uma *flag* que indica, se as medidas são necessárias no momento, ou se é possível esperar por elas até que estejam disponíveis. Deste modo, são implementados os dois modos necessários de funcionamento: síncrono para quando for necessário obter amostras imediatamente, ou assíncrono, se for aceitável esperar por eventos dos sensores. Estrutura de uma instrução de recolha de dados é visível na figura 4.15.

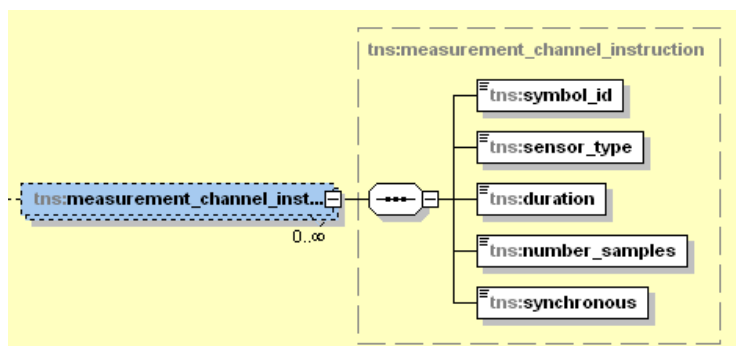


Figura 4.15: *Schema* da instrução de obtenção de um conjunto de amostras

**Realizar análises** Instruções deste tipo, permitem realizar análises sobre dados. O conceito de análise neste sistema corresponde uma *caixa negra*, que recebe um *input* e *output*, independentemente do que estes sejam. As análises, são objectos com uma *interface* conhecida, que estão acessíveis no sistema através de catálogos, que são invocados nestas instruções. Esta estrutura é visível na figura 4.16.

**Regras** Instruções deste tipo, permitem aplicar operadores binários, *booleanos* ou lógicos, sobre outros resultados já obtidos durante a execução das regras do Padrão de Observação.

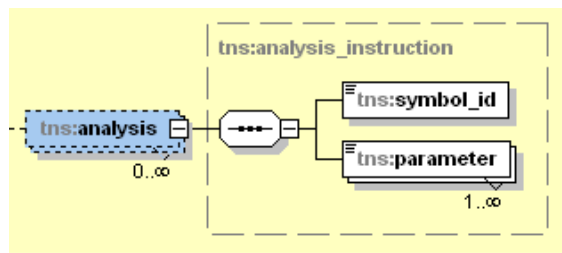


Figura 4.16: *Schema* da instrução de realização de uma análise

Os resultados guardados por estas regras são resultados *booleanos* e correspondem às condições que importa verificar no decorrer da execução da aplicação. A estrutura de uma regra é visível na figura 4.17.

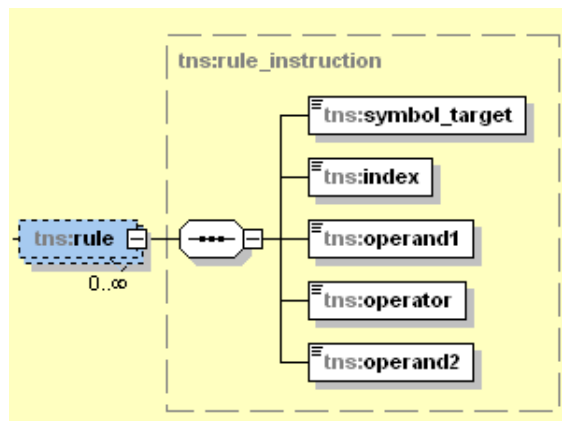


Figura 4.17: *Schema* da instrução de aplicação de uma condição

**Instruções de salto condicionais** As instruções de um Padrão de Observação estão divididas em blocos de execução. Um bloco tem várias instruções. Uma instrução de salto condicional, permite saltar entre blocos de instruções, se verificar uma dada condição. Isto é visível na figura 4.18.

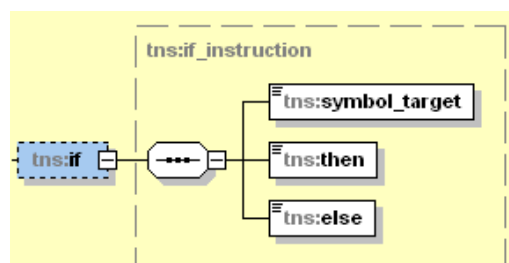


Figura 4.18: *Schema* da instrução de salto condicional

**Conclusões a retirar sobre os dados** As conclusões, são as instruções finais de um Padrão de Observação. As instruções de conclusão, estabelecem o resultado final atribuindo uma prioridade ao estado de saúde do utilizador e um diagnóstico. As conclusões, são tiradas com base num resultado de uma regra, que pode, por sua vez, ter resultado de um longo encadeamento de regras, de recolhas de dados e de análises desses dados. A dimensão temporal, é incluída nas instruções de conclusão através de períodos de tempo, nos quais uma determinada condição tem de ser monitorizada e verificada positivamente. Isto é visível na figura 4.19.

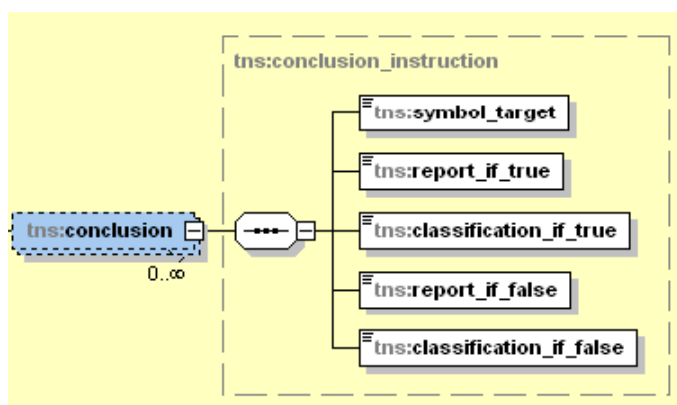


Figura 4.19: *Schema* da instrução de conclusão

#### 4.3.5.6 Interfaces com Sistemas Externos

Internamente, cada sistema externo vai ser representado, por um componente interno dentro do sistema, implementando desta forma o padrão *Adapter*.

O sistema externo, nesta instância do sistema é o servidor que contém toda a informação médica. O componente descrito ao longo desta secção, sincroniza automaticamente a informação com esse servidor e a comunicação é realizada através de mensagens. O comportamento do sistema é visível na figura 4.20.

As mensagens com o componente externo, têm como finalidade a troca da informação médica, e todo o *profile* do utilizador. Esta informação médica é composta por:

- Padrões de Observação, para a recolha dos dados e raciocínio sobre os mesmos
- Agendas.
- Proposições lógicas que permitem efectuar o diagnóstico, a partir dos dados existentes.

A informação partilhada, não se resume no entanto, à obtenção do *profile* do utilizador e consiste também, no envio da informação médica que for sendo recolhida pela plataforma médica, quer tenha sido obtida por invocação periódica, ou por invocação devido a um evento médico.

## Sistema Pessoal de Saúde com Capacidades de Diagnóstico

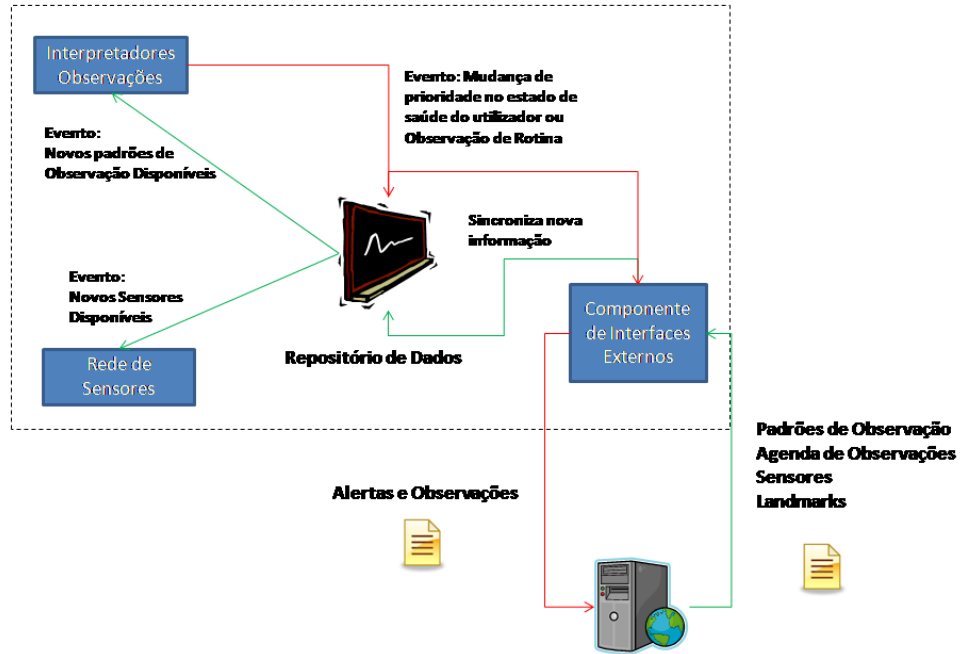


Figura 4.20: Interfaces entre os componentes internos e o servidor externo

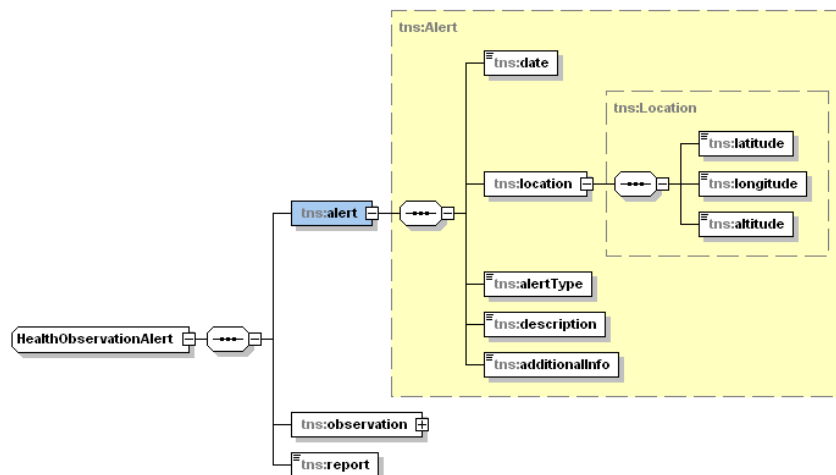


Figura 4.21: *Schema* de alerta com observação

Esta informação partilhada, consiste em Observações médicas (secção 4.3.4), com o posicionamento georeferenciado do utilizador, ou alertas indiciando uma situação de emergência médica. Esta estrutura é visível nas figuras 4.21 e 4.22.

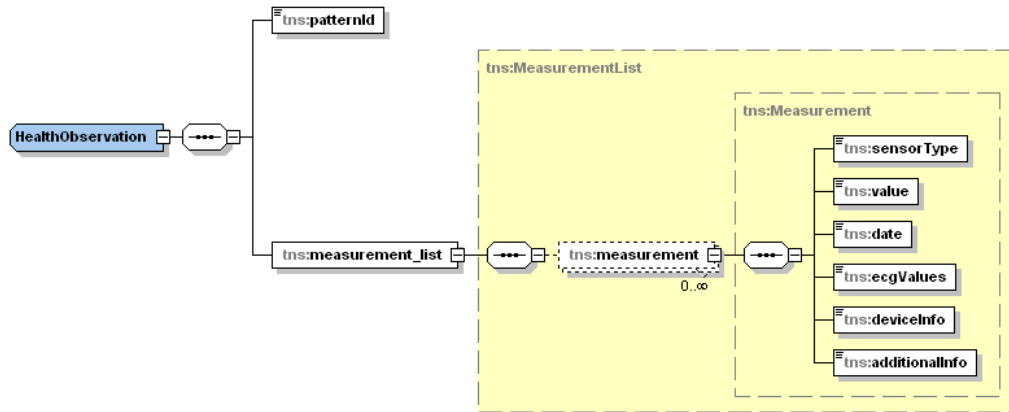


Figura 4.22: *Schema* de uma observação

## 4.4 Arquitectura Física

### 4.4.1 Integração

#### 4.4.1.1 Integração com o Sítio Web

A tecnologia de *Web Services*, foi utilizada para estabelecer a interacção com o *site* para onde são enviados os alertas e de onde é recolhida toda a informação de contexto. A tecnologia *Web Services*, baseia-se no envio de mensagens do protocolo *Simple Object Access Protocol (SOAP)*.

O protocolo *SOAP*, permite a troca de objectos remotamente, recorrendo à linguagem *XML* para os codificar. A tecnologia *Web Services* é um *standard* da indústria, largamente utilizado onde quer que seja necessária a integração de dois sistemas de *software*. A sua principal vantagem em termos de arquitectura, é o grau de independência que permite entre os dois sistemas a integrar. Existem também, uma grande quantidade ferramentas de desenvolvimento para esta tecnologia, o que permite acelerar e muito o desenvolvimento, de sistemas baseados nesta tecnologia.

A *Web Services Description Language (WSDL)*, é uma linguagem *XML* adoptada pela *W3C* como padrão e serve para descrever o interface dos métodos fornecidos por um *Web Service*, bem como o retorno e parâmetros de cada um destes métodos. Tanto os parâmetros, como o retorno dos métodos podem ser objectos ou conjuntos de objectos, que têm ser definidos através das normas *XSD schema*. Com as tecnologias actuais, utilizadas para *Web Services*, é possível criar um *WSDL* e respectivos schemas adicionais descrevendo toda a estruturação dos objectos em *XML*, a partir das classes dos objectos envolvidos e o contrário também é possível.

#### 4.4.1.2 Conectividade com os Sensores

Para a conectividade com os sensores, foi utilizada a *Bluetooth Wireless Technology*, devido ao facto de ser uma tecnologia com reduzido custo energético e de ser compatível com o *hardware* utilizado. Trata-se de uma tecnologia de comunicação sem fios, para pequenas distâncias, e o consórcio criador da tecnologia pretende que seja um *standard* para a comunicação entre dispositivos electrónicos. Existem várias configurações possíveis na utilização da tecnologia *Bluetooth*, nomeadamente, *piconets* e *scatternets* e existem dois papéis possíveis para um dispositivo quando está presente numa destas redes: *master* e *slave*. Uma rede *piconet*, pode ir até 8 dispositivos em comunicação, nos quais 1 deles tem o papel de *master*, e tem como funcionalidades definir as portadoras a utilizar pelos outros dispositivos. Nas redes *scatternet*, um dos dispositivos alterna entre o papel de *slave* numa *piconet* e o papel de *master* noutra *piconet*, permitindo assim estender o numero de dispositivos ligados em rede.

Neste sistema, foi utilizada uma configuração *piconet*, em que o *Smartphone* desempenha o papel de *master*. Foi utilizado o perfil *Bluetooth* de *Serial Port Profile*, que permite emular portas de série sobre a comunicação *Bluetooth*.

#### 4.4.2 Aplicação Móvel

A aplicação móvel, foi implementada com recurso à tecnologia *J2ME*, a um conjunto de bibliotecas e às *API's standard*, implementadas pelo dispositivo escolhido para o desenvolvimento do protótipo. A visão geral da arquitectura física pode ser vista na figura 4.23.

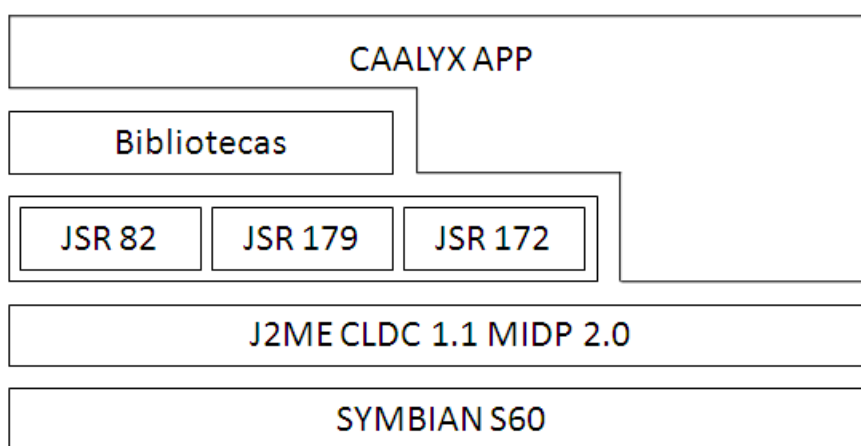


Figura 4.23: Arquitectura Física da Aplicação Móvel

#### 4.4.3 Java para Dispositivos Móveis

A tecnologia *Java for Mobile (J2ME)* da *Sun Microsystems*, foi utilizada na implementação da plataforma de *software*. A principal razão da utilização desta tecnologia, foi a portabilidade que esta permite entre dispositivos. O perfil *J2ME* utilizado, foi o *CLDC 1.1* e as bibliotecas *MIDP 2.0*.

**Java Specification Requests** Os *Java Specification Requests* são especificações de *software*, que são geridas e evoluídas pelo *Java Community Processes*. Existem especificações para muitos e variados fins, como definir um conjunto *standard* de classes e métodos para utilizar a tecnologia *bluetooth*, definir toda a arquitectura da biblioteca *MIDP* ou definir todas as classes e *interface* de uma *framework OSGI* para *dispodream theater change of seasons*ssitivos móveis. Estes *Java Specification Requests*, têm como objectivo servir de *standard* para todos os fabricantes que quiserem implementar bibliotecas ou *frameworks*, destinadas aos fins cobertos pelos *Java Specification Requests*.

**JSR-82 – API de Bluetooth** Esta especificação, define todas as classes e interfaces que devem existir numa *API* para aceder aos vários protocolos e camadas da *Bluetooth Protocol Stack*. São acessíveis os protocolos *L2CAP*, o perfil *Serial Port Profile* e os Serviços de Descoberta, bem como algumas funcionalidades de segurança, como por exemplo, a autenticação de ligações.

**JSR-172 - JAX-RPC** A *JSR 172*, é a especificação *J2ME* existente para utilização de *Web Services*. Esta *API*, permite a desserialização dos objectos a partir do *XML* presente nas Mensagens *SOAP*, evitando assim, a necessidade de implementar esta transformação. Para tal, basta que com o auxílio de ferramentas se gerem os *stubs* a partir do *WSDL* do *Web-Service*. No entanto, nem todas as construções gramaticais feitas em *XML* são convertíveis, devido às limitações desta *API* em run-time.

**JSR-179 – API de Localização** Esta especificação refere-se à *API* de georeferenciação para dispositivos móveis. Permite o acesso a todas as funcionalidades mais comuns de geoposicionamento, tais como obter a posição georeferenciada num dado momento, obter a orientação do dispositivo em relação ao norte verdadeiro, ou obter a velocidade a que viaja o dispositivo num dado momento. Esta *API* não distingue qual a fonte de informação a ser utilizada e como tal, tem de ser o fabricante na sua implementação deste *standard*, a definir que fonte de informação deve ser utilizada.

## 4.5 Pormenores de Implementação

### 4.5.1 Implementação da Comunicação Assíncrona entre Componentes

A arquitectura orientada aos eventos, foi implementada recorrendo basicamente a dois tipos de componentes, nomeadamente, ao *middleware* e aos *Observers*. Os *Observers*, são a base de todos os componentes da aplicação e são capazes de enviar e receber mensagens sobre determinados assuntos, de modo completamente assíncrono e concorrente. Isto corresponde à implementação do padrão *Observer*. Ao mesmo tempo, cada mensagem define as acções a realizar no receptor, o que constitui a implementação do padrão *Command*. Um *Observer*, corre na sua própria *thread* e vive permanentemente num ciclo de processar e esperar por novas mensagens. Para evitar perdas de mensagens, todos os *Observers* têm uma *queue* onde são guardadas as mensagens até serem processadas. Isto é visível na figura 4.24.

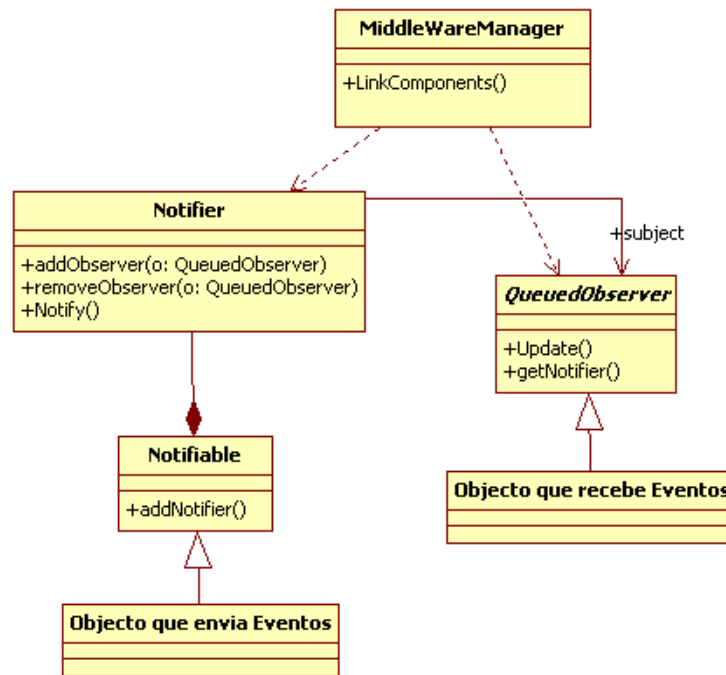


Figura 4.24: Implementação do Padrão *Observer*

Os *Observers* são associados ao *Repositório* ficando deste modo, associados indiretamente entre eles.

#### 4.5.2 Persistência dos Dados

A persistência dos dados é um problema complexo, visto que não existem bases de dados disponíveis para este perfil *J2ME* e ao mesmo tempo, também não existe o mecanismo de *reflection*. No *J2ME* existe apenas o *Java RMS*, que permite guardar conjuntos de *bytes* na memória do telemóvel, o que é extremamente lento. Implementar um mecanismo destes manualmente, implicaria a criação de *parsers* e de todos os mecanismos de sincronização, o que seria um esforço de implementação adicional bastante elevado.

A *framework floggy* é uma ferramenta *open source* de persistência de objectos, para o perfil *CLDC* da tecnologia *J2ME*. Esta *framework*, fornece primitivas simples que permitem salvar, editar e remover objectos, utilizando a tecnologia de registos *java RMS*. Esta tecnologia, foi escolhida devido à escassez de opções com este tipo de funcionalidades.

#### 4.5.3 Implementação do Interpretador de Observações

Um padrão de observação, irá corresponder a uma sequência de instruções, que se-guem a especificação da linguagem de observação proposta na secção 4.3.5.5.

Cada padrão de observação, corre no seu próprio interpretador e cada ciclo de vida da aplicação CAALYX terá várias instâncias de interpretadores de observação activas. Cada interpretador de observação, tem um *container*, onde são guardadas todas as suas variáveis, nomeadamente condições, resultados de obtenção de amostras, resultados de análises e resultados da aplicação de regras. Só este interpretador tem acesso ao seu próprio *container* de informação. Isto é visível na figura 4.25.

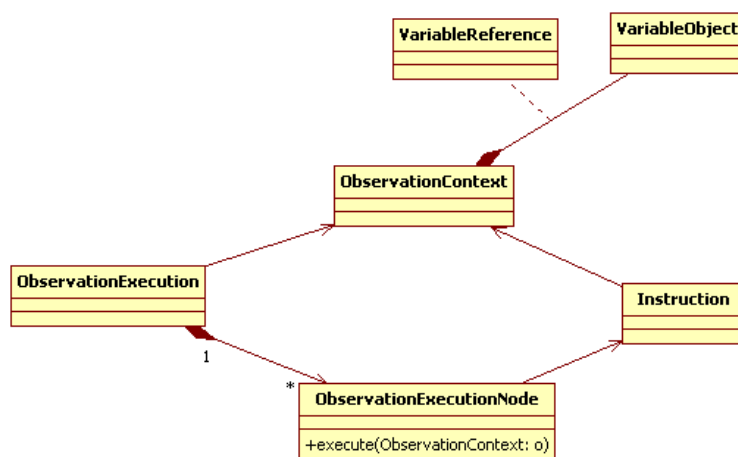


Figura 4.25: Estrutura estática do interpretador

Em termos de implementação, embora o interpretador constitua uma implementação complexa, dado a robustez necessária face à grande quantidade de *inputs* possíveis, considera-se que existem dois desafios que mereceram mais atenção, que são a obtenção de amostras e a implementação das instruções que utilizem a variável temporal.

#### 4.5.3.1 Obtenção de Amostras

A obtenção das amostras, foi um dos maiores problemas enfrentados na aplicação, visto que nem sempre, as amostras estão disponíveis no momento da execução e nestes casos, será necessário obter novas medidas a partir dos sensores. Devido ao facto de existirem vários interpretadores, com diferentes padrões de observação a serem executados concorrentemente, interagir directamente com os sensores, poder-se-ia revelar muito problemático. Assim, para resolver este problema, utilizou-se o *theater change of seasons* o repositório de dados como *cache*. Um interpretador de observações tenta obter as amostras primeiramente junto do repositório de dados. Apenas se não existirem amostras suficientemente recentes no repositório, capazes de suprimir as necessidades, novas medidas serão obtidas com a ajuda da rede de sensores. Isto é visível na sequência da figura 4.26.

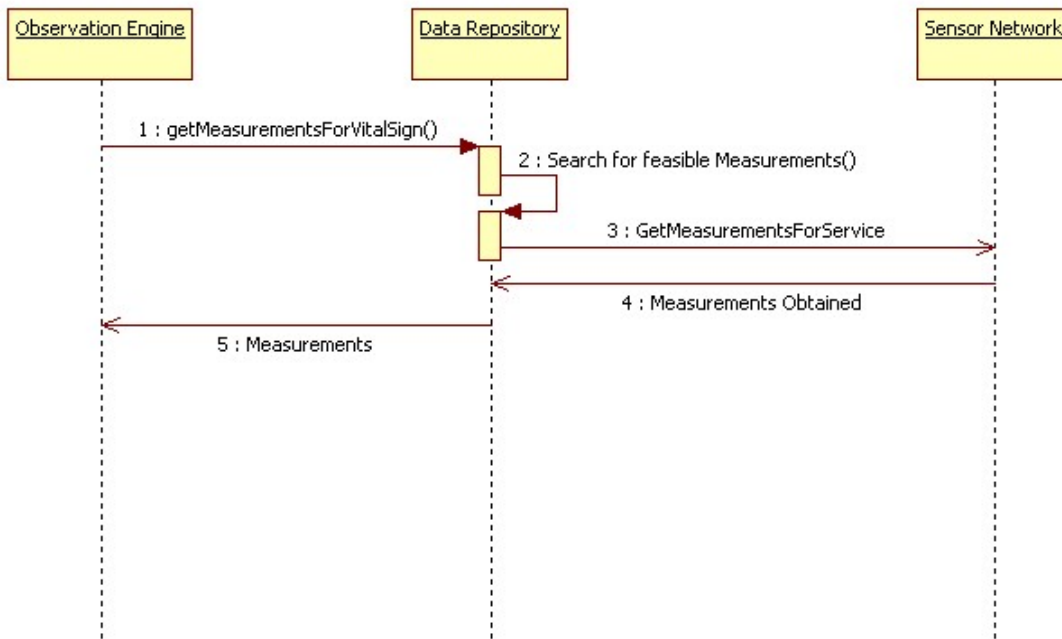


Figura 4.26: Diagrama de Sequência para obter Medidas

#### 4.5.3.2 Dimensão temporal

A dimensão temporal, existe em variados casos de processos de diagnósticos. A dimensão temporal, foi incluída na linguagem de observação, através de dois tipos de instruções, instruções de espera e conclusões temporais.

As instruções de espera, permitem fazer pausas no fluxo de execução. Isto é útil, em processos de diagnóstico, em que seja necessário verificar uma condição num instante de tempo, esperar alguns instantes, recolher amostras e verificar as condições novamente.

As conclusões temporais, permitem a determinação de períodos para verificar determinadas condições ao longo do tempo. Ao fim desses períodos de tempo, se a condição se mantiver, é lançado o alerta tal como nas outras conclusões normais. Para avaliar estas condições, serão lançadas *threads* que acedem ao *container* de informação do interpretador, para verificar se as condições se alteraram ou não.

#### 4.5.4 Localização

O geoposicionamento, foi conseguido através da utilização do *GPS* interno do *Smartphone* e da utilização da *API JSR-179* de localização. A funcionalidade pretendida, necessitava de um cálculo simples da distância entre dois pontos, de modo a poder calcular se o idoso se encontra dentro dos limites pretendidos, em determinados pontos de referência. Isto poderá ser útil para idosos com algum tipo de perturbações cognitivas. As coordenadas

geográficas, são necessariamente dadas sobre um referencial e existem vários referenciais possíveis. As coordenadas *GPS* são obtidas sobre o referencial *WGS84*, que se trata de um referencial elipsoidal. Isto significa, que a distância entre dois pontos marcados neste referencial, não é equivalente à distância num referencial cartesiano. O cálculo da distância nesta situação, pode ser calculada de duas maneiras: através da projecção de coordenadas, ou através da menor curva elipsoidal entre dois pontos. A tarefa de projecção de coordenadas é uma tarefa largamente utilizada em *Sistemas de Informação Geográfica*, porque permite que a seguir sejam utilizadas as ferramentas de cálculo para referenciais cartesianos, no entanto, é uma tarefa extremamente exigente computacionalmente, portanto não foi possível utiliza-la no dispositivo escolhido neste protótipo. O cálculo do comprimento da menor curva elipsoidal, foi realizada através da biblioteca *Gavaghan* desenvolvida originalmente para *geocaching*. Esta biblioteca é *Open Source* e foi desenvolvida originalmente para a plataforma *J2SE*, e teve de ser adaptada para a plataforma utilizada neste protótipo.

## Capítulo 5

# Experimentação

Toda a parte de experimentação do projecto, foi realizada utilizando como dispositivo de desenvolvimento, o *Nokia N95*. Devido ao facto de ainda não se ter realizado um teste global do sistema, (este será realizado no segundo ano de projecto) a experimentação do sistema, foi realizada recorrendo a testes de integração, de modo a testar os vários aspectos do sistema assim como uma experiência de âmbito mais reduzido, realizada dentro do projecto CAALYX. Este capítulo, apresenta os testes realizados em cada um dos aspectos de modo a demonstrar o estado de conclusão do projecto.

### 5.1 Integração com Sensores

#### 5.1.1 Desenvolvimento

Durante o desenvolvimento do projecto, integraram-se diferentes sensores médicos na aplicação CAALYX. Foram integrados, um sensor de pressão arterial, um sensor de detecção de quedas e um electrocardiografo portátil. O sensor de quedas, foi desenvolvido no âmbito do projecto CAALYX pelos parceiros da Universidade de *Limerick*. A integração implicou a criação de um *driver*, de *configuradores* e de *descodificadores* para os dados médicos obtidos dos sensores.

Tirando o necessário esforço de desenvolvimento destes artefactos, estes não implicaram alterações à estrutura da rede de sensores, o que validou a estrutura utilizada. Todos os sensores tinham protocolos diferentes, embora comportamentos similares, o que permitiu que as abstrações utilizadas funcionassem com todos os sensores. O sensor de quedas e o sensor de pressão arterial, funcionavam em modo autónomo, o que significa, que conseguiam iniciar a comunicação sempre que algo relevante acontece e o electrocardiografo era um sensor *slave*, que não tinha autonomia para iniciar a comunicação.

Encontra-se também em desenvolvimento pelo consórcio, um *Wearable Light Device* que permite a integração de vários sinais vitais e que pode funcionar em diferentes modos:

suporta pedidos de obtenção de dados e funciona em modo assíncrono. Este sensor, suporta também a reconfiguração dinâmica dos valores limite para os sinais vitais, de modo a reconfigurar quais os eventos que devem ser reportados e para que valores limite estes devem ser reportados.

A integração deste sensor, foi mais complexa que os anteriores. Este sensor, tem um modo de operação diferente e como tal, foi necessário mudar os mecanismos da rede de sensores e as abstrações dos sensores tiveram de ser melhoradas para abranger completamente as novas funcionalidades de reconfiguração e descodificação. Este sensor, não foi ainda integrado devido ao facto, de não estar acabado à data de escrita desta dissertação.

### 5.1.2 Funcionalidades de Gestão da Rede de Sensores

**Registo, Detecção e Remoção de Sensores** As funções de registo, detecção e remoção de sensores, manipulam os dados do repositório, de modo a alterar a configuração e o comportamento de sensores. Estas funções, podem ser realizadas manualmente pelo administrador do sistema, ou automaticamente através da integração com o servidor dessa informação. A figura 5.1 retrata as funcionalidades de gestão da rede de sensores, em modo manual.



Figura 5.1: Funcionalidades de Registo de Sensores

**Leitura dos Sensores** Foram implementadas as funções de leitura dos três sensores inscritos na *BAN*, e todas funcionaram correctamente, sendo possível ler dos três sensores com fiabilidade.

## Experimentação



Figura 5.2: Execução da leitura dos sensores

### 5.1.3 Integração dos Processos de Diagnóstico

Para que o *CAALYX* possa funcionar, é necessário a existência de conhecimento que possa ser utilizado na aplicação. Por outras palavras, é necessária a existência das árvores de decisão médicas.

|  | Units            | Absolute low | Absolute high | Relative Low | Relative High I | Relative High II |
|--|------------------|--------------|---------------|--------------|-----------------|------------------|
| Heart Ratio  | beats per minute | 44           | 120           | 45-55        | 97-119          | none             |
| Respiratory Rate   | resp. per minute | none         | 35            | none         | 27-34           | none             |
| Temperature  | ° celsius        | none         | 38            | none         | 37.1 - 37.4     | 37.5 - 37.9      |
| These limits are applicable to elders at rest: no motion in 5-10 minutes (5 minutes may be enough) |                  |              |               |              |                 |                  |
| NO heart or lung diseases (we will provide later on other limits for such diseases)                |                  |              |               |              |                 |                  |
| No drugs affecting heart ratio (we will provide later on other limits for such drugs)              |                  |              |               |              |                 |                  |

Figura 5.3: Thresholds para os sinais vitais

O projecto *CAALYX* é um projecto multidisciplinar e como tal, tem parceiros com conhecimento na área da medicina, que forneceram estas árvores de decisão. As árvores de decisão integradas são visíveis nas figuras 5.3 e 5.4.

Foi necessário codificar este conhecimento, na linguagem descrita na secção 4.3.5.5, o que serviu também para validar a linguagem e experimentá-la com conhecimento médico real. Foi possível codificar as árvores de decisão na linguagem pretendida o que validou a linguagem. O único senão verificado foi o tamanho dos ficheiros *XML* resultantes que se tornaram verdadeiramente extensos.

## Experimentação

| Temperature II<br>High: 37.5 - 37.9 | Heart Ratio | Respiratory Ratio | Action 1 | Action 2    | Action 3  | Action 4    |
|-------------------------------------|-------------|-------------------|----------|-------------|---|-------------|
| Normal                              | Normal      | Normal            | none     |             |   |             |
| High                                | Normal      | Normal            | 10'      | 30'         | FEVER   |             |
| High                                | High        | Normal            | FEVER    |             |   |             |
| High                                | Normal      | High              | FEVER    |             |   |             |
| High                                | High        | High              | FEVER    |             |   |             |
| Normal                              | High        | Normal            | 30'      | 30'         | TACHYCARDIA   |             |
| Normal                              | High        | High              | 10'      | TACHYCARDIA |   |             |
| Normal                              | Normal      | High              | 30'      | 30'         | TACHYPNEA   |             |
| Normal                              | Low         | Normal            | 15'      | 15'         | 15'   | BRADICARDIA |
| Temperature I<br>High: 37.1 - 37.4  | Heart Ratio | Respiratory Ratio | Action 1 | Action 2    | Action 3  |             |
| High                                | Normal      | Normal            | 30'      | 30'         | repeat every 1 hour (inform after in 6 hours if persistent) |             |
| High                                | High        | Normal            | 30'      | 30'         | ALARM   |             |
| High                                | Normal      | High              | 30'      | 30'         | ALARM   |             |
| High                                | High        | High              | FEVER    |             |   |             |

Figura 5.4: Execução da leitura dos sensores

### 5.1.4 Integração com Sítio Web e Integração da Informação Médica

A gestão do *profile* do utilizador, consistia fundamentalmente na integração com o servidor através de *Web-Services*. As funcionalidades deste *package*, seriam o *download* e integração dos padrões de observação e o *download* das descrições dos sensores e integração destes sensores na rede. Seriam também o envio de observações e de alertas reportando, situações anómalas.

**Download da Informação médica** O *download* da informação médica, consistia na recepção dos padrões de observação, sensores e posicionamentos. O principal problema, residia na recepção do padrão de observação e a sua integração com o repositório de dados. Este problema foi resolvido, e é possível integrar os padrões de observação com os dados do repositório. Foram testados diversos padrões de observação, na integração com o sistema. Na figura 5.5, é possível visualizar os padrões de observação e os seus valores limite integrados com a aplicação móvel.

## Experimentação

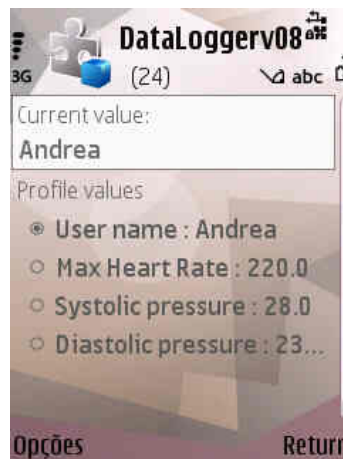


Figura 5.5: Integração dos thresholds

**Envio da Informação Médica** O envio da observação médica, foi cumprido, quer no envio de alertas e no envio de observações médicas, com os respectivos dados médicos associados. O envio de informação é visível nas figuras 5.6 e 5.7.

CAALYX

Health  
Complete Ambient Assisted Living Experiment

European Union

Caalyx > Home > View History Elders Alerts

User: dottore

Options Menu

- Alert
  - View History Elders Alerts
- Observation Pattern
  - Add Observation Pattern
  - Look for Observation Patterns
  - Modify Observation Pattern
  - Delete Observation Pattern
- Measurement
  - View History Elders Measures

The alert has the following data:

Elder: Pepino      Alert Type: ATTENTION  
Time: 30/05/2008 - 18:46:40      State: ACTIVE  
Description: Out of Threshold Observation  
Additional Info: no additional info  
Location: [Map]

Health Alert

Report: No Report Available

| LIST OF MEASUREMENT OF ELDER:           |                             |
|---|-----------------------------|
| Sensor Type: ECG_SENSOR<br>Value: null  | Time: 30/05/2008 - 18:46:39 |
| Sensor Type: HEARTRATE<br>Value: 94 bpm | Time: 30/05/2008 - 18:46:39 |
| Sensor Type: HEARTRATE<br>Value: 91 bpm | Time: 30/05/2008 - 18:46:39 |

Copyright 2007. CaretakerSite Web | Legal warning and conditions of use

Figura 5.6: Envio de uma Observação

## Experimentação



Figura 5.7: Envio de electrocardiograma

### 5.1.5 Integração com Geoposicionamento

Outro dos contextos, é o contexto geográfico, que consiste em ver se o geoposicionamento do utilizador está mais afastado que uma determinada tolerância descrita no *profile* do utilizador e se isto acontecer, este geoposicionamento deve ser enviado para o servidor de contexto. Foram realizados alguns testes em cenário real utilizando o *GPS*. Foram verificadas algumas falhas, sobretudo no próprio dispositivo de *GPS*, embora sempre que foi possível obter uma posição válida do *GPS*, os algoritmos funcionaram convenientemente e enviaram para o servidor de contexto este geoposicionamento, tal como era esperado.

### 5.1.6 Integração Longitudinal com Diagnóstico

A integração longitudinal com o Diagnóstico, consistia nos testes de funcionamento do sistema, através da sua utilização esperada. O sistema deveria integrar o conhecimento automaticamente e realizar diagnósticos, tal como especificado ao longo do capítulo 4.

Os testes realizados, foram pequenos testes de integração com padrões de observação simples, de modo a poder avaliar o funcionamento do sistema. O maior teste realizado, consistiu na integração de um padrão de observação que era formado por uma sequência de instruções, que permitiam esperar por um evento do medidor de pressão arterial, recolher automaticamente um electrocardiograma que durava um minuto. Depois, através de um conjunto de *thresholds* simples, decidir pelo envio de um alerta ou não. Todo o teste funcionou de forma correcta, desde a integração dos dados dos padrões de observação,

até ao envio da observação para o sítio onde foi integrada a informação médica. Este foi o único teste significativo realizado.

## 5.2 Ensaio da Primeira Iteração do Projecto CAALYX

Como projecto europeu, o *CAALYX* foi sujeito a um ensaio oficial para validação do estado do projecto. O *CAALYX* é direccionado para a população idosa e como tal, participaram na experiência, um conjunto de idosos. Este ensaio, foi realizado em Itália, em Outubro de 2007. O objectivo destes testes oficiais, era a recolha de dados relativamente a todo o sistema, de modo a avaliar o seu modo de funcionamento e validar as suas funcionalidades perante os utilizadores finais. Participaram também na experiência profissionais de saúde, para avaliar o sistema do ponto de vista de usabilidade. Mais concretamente, os objectivos deste ensaio foram os seguintes:

- Obter comentários dos utilizadores sobre a usabilidade do sistema.
- Obter comentários do pessoal que presta cuidados médicos.
- Testar numa perspectiva de desenvolvimento as tecnologia e a arquitectura adoptada, numa situação real.
- Compreender melhor as implicações da adopção do sistema *CAALYX*.
- Refinar a especificação de requisitos para a segunda iteração do projecto.
- Determinar se a monitorização realizada, afecta os padrões normais de mobilidade dos idosos.

### 5.2.1 Configuração

O *hardware* utilizado constitui-se por um *Smartphone* e por um conjunto de sensores médicos. O *Smartphone*, é a plataforma móvel responsável pela interface com os sensores e com o servidor de contexto e por todo o processamento necessário. O modelo utilizado para este *Smartphone*, foi um *N95* da *Nokia*. Este modelo contém um bom leque de tecnologias de conectividade, como *Bluetooth*, *Wifi* e acesso à *Internet* através dos operadores, o que cumpre todos os requisitos de conectividade para o desenvolvimento do protótipo. Possui ainda, um módulo *GPS*, o que permite resolver também os problemas relacionados com o geoposicionamento.

### 5.2.2 Descrição do Ensaio

O ensaio realizou-se em *Ancona*, Itália com idosos residentes no lar da terceira idade. A experiência realizou-se na enfermaria do lar, e na própria cidade de *Ancona*, por onde os idosos transportaram o sistema *CAALYX*.

Participaram na experiência dez idosos, um operador e profissionais de saúde que trabalham no lar *Benincasa*. Estes, avaliaram os dados fornecidos pelo sistema e ajudaram os idosos na sua utilização. Os ensaios realizaram-se ao longo de dois períodos de duas semanas, com dois grupos de 5 idosos em cada um dos períodos.

O protótipo utilizado na experiência, consistia numa plataforma móvel, com o *software* desenvolvido a interagir com um sensor de pressão arterial, um electrocardiógrafo portátil e um sensor de quedas, desenvolvido também no âmbito deste projecto.

Durante vários dias da semana, cada idoso utilizava os sensores de pressão arterial e electrocardiógrafo, de modo a recolher dados que eram enviados para o servidor de dados médicos ainda sem diagnóstico associado, mas com a recolha orquestrada por um padrão de observação. Esta recolha, consistia em três medidas de pressão arterial e um electrocardiograma. Estes dados, eram enviados para o sistema central, onde eram armazenados. Seguidamente, os idosos transportavam o sensor de quedas durante 8 horas por dia, que em caso de detecção de quedas, estas eram reportadas ao servidor, em tempo real, através do dispositivo móvel.

### 5.2.3 Resultados Obtidos

Para avaliar os resultados, foram propostos questionários aos utilizadores e profissionais de saúde que participaram na experiência, assim como entrevistas de resposta aberta, de modo a obter resultados qualitativos relativos ao sistema.

#### 5.2.3.1 Resultados Qualitativos

Relativamente à utilização do sistema móvel tanto, utilizadores, como profissionais de saúde, salientaram que o sistema tinha potencialidades e que a sua maior potencialidade era sem dúvida, a capacidade de enviar alertas autonomamente e a sua ubiquidade.

Comentários dos utilizadores, salientaram particularmente a utilidade do sistema quando eles se encontram sozinhos e sem possibilidade de pedir ajuda. Algumas destas situações, são as quedas, que são muito comuns na população idosa e são muitas vezes incapacitantes. O protótipo desenvolvido possibilitava o envio de pedidos de ajuda nestas situações. Outros dos comentários, foram relativos à dificuldade nas tarefas de administração da rede de sensores do dispositivo móvel.

Comentários dos profissionais de saúde que ajudaram na gestão do sistema, salientaram o facto de que terem um dispositivo electrónico, capaz de os alertar em situações

de emergência, seria extremamente útil, visto que eles não eram capazes de controlar um grande número de idosos.

Em geral o sistema teve poucas falhas durante o funcionamento do ensaio e as falhas verificadas, foram sobretudo no *interface* com os sensores, ou falhas humanas causadas por alguma incompreensão do sistema quando foi necessária a sua configuração manual.

## 5.3 Discussão dos Resultados

### 5.3.1 Diagnóstico Adaptável

A abordagem utilizada é completa, no sentido de que cobre todas as situações de detecção necessárias, ou pelo menos tão completa como qualquer outra abordagem baseada em conhecimento, que se limite a deduzir logicamente sobre eventos de mudança de contexto, como a utilização de ontologias. Esta abordagem, também permite a obtenção de novos dados, se estes forem necessários no processo de diagnóstico, não se limitando a dados obtidos por eventos.

As limitações da solução, devem-se ao facto de que há a possibilidade de perda de eventos, em casos nos quais tenha de ser a aplicação a despoletar a recolha de dados. Para estes casos, existe no entanto, a possibilidade da existência de uma agenda com uma frequência pré-especificada, ou até mesmo a recolha e análise permanente de dados de um sinal vital. Segundo pareceres médicos, na esmagadora maioria das vezes, não é necessário recolher dados permanentemente e será apenas necessário, uma agenda.

### 5.3.2 Linguagem de Diagnóstico

A linguagem utilizada foi baseada no *standard* da *OGC Observations and Measurements*, e como parte de um modelo de objectos adaptável. Conceptualmente, a solução funcionou bem e cumpriu todos os objectivos que se pretendiam de adaptabilidade. Permitiu a integração de vários padrões médicos, que foram codificados nesta linguagem, e diferentes contextos de actividade física e agendas.

A codificação nesta linguagem revelou-se no entanto, um processo longo e demorado, dado o número de combinações de combinações, que foi necessário codificar. Isto indica, que o *encoding* utilizado não foi o mais apropriado. Das várias alternativas, seria melhor a utilização de uma linguagem mais declarativa e se possível, *standard* da indústria, tal como *Arden Syntax* da *HL7*.

### 5.3.3 Arquitectura utilizada

A arquitectura utilizada, foi uma arquitectura *blackboard*, que funciona sobretudo com base em eventos. Com esta abordagem para a arquitectura escolhida, ganhou-se flexibilidade na construção de protótipo, havendo a possibilidade de adicionar e remover componentes da aplicação da maneira mais simples possível durante o desenvolvimento, devido à utilização do estilo de *loosely coupled components*.

No entanto, isto tornou a implementação mais longa, devido ao facto de se ter de implementar um esquema de passagem de mensagens em componentes completamente concorrentes, e da resolução de todos os problemas de concorrência que podem existir. Conseguiu-se no entanto, uma aplicação robusta e bastante flexível, que facilmente pode evoluir para uma *framework*.

### 5.3.4 Tecnologias utilizadas

As tecnologias utilizadas para o desenvolvimento do protótipo, revelaram-se adequadas, embora se tenham verificado algumas limitações tecnológicas, principalmente ao nível do *Bluetooth*. A tecnologia *J2ME*, trouxe algumas vantagens, tais como a portabilidade da aplicação, no entanto, trouxe muitas restrições do nível do controlo da aplicação, devido às políticas de segurança presentes nos dispositivos móveis utilizados no protótipo. As restrições tecnológicas da plataforma *J2ME*, também se revelaram um problema no desenho da aplicação, como a ausência do mecanismo de *reflection* e o facto da *API* de *Web Services* não permitir algumas das construções gramaticais do *xsd: schema*, o que obrigou a simplificar a linguagem de observações.

Ao nível de persistência, a *J2ME* revelou-se muito ineficiente para guardar dados, o que implicou que determinados dados não sejam persistentes, ficando dependente da obtenção dos dados através de *Web Services*, com as devidas consequências.

A integração dos *Web Services*, revelou-se uma ótima solução, simplificando muito a *interface* entre os dois sistemas, fornecendo uma *interface padronizada*, num paradigma *loose coupled*. O único senão dessa abordagem, é provavelmente, a transferência de grandes quantidades de informação. De acordo com os resultados, verificaram-se alguns problemas para quantidades de dados maiores, embora se considere que este possa ser um problema de implementação e não do uso da tecnologia em si.

## 5.4 Discussão dos Resultados Obtidos

Dos resultados obtidos, pode-se dizer que cada um dos aspectos fundamentais da integração em separado está resolvido, referindo-se neste caso à Integração com os sensores, à integração com o sítio que contém a informação médica e ao posicionamento georeferenciado. O diagnóstico, foi utilizado com padrões de observações muito simples

## Experimentação

e não se pôde comprovar o funcionamento integrado do sistema, com contextos físicos e com agendas e com casos de padrões de observação complexos, em ambiente real. A linguagem dos padrões de observação, foi integrada com conhecimento médico real o que validou a linguagem. O ensaio geral do sistema, permitiu obter comentários positivos dos utilizadores em relação à utilidade do mesmo, embora não se tenha podido testar as funcionalidades de diagnóstico integradas. Apesar de ter falhas, o desempenho nestes ensaios foi positivo e considera-se que valida os princípios utilizados para a integração de todos os componentes.

## Capítulo 6

# Conclusões e Trabalho Futuro

Este capítulo, apresenta as principais contribuições dadas por este trabalho e as linhas orientadoras para trabalho futuro.

### 6.1 Sumário das contribuições

Esta dissertação, descreveu um trabalho realizado sobre a aplicação da sensibilidade ao contexto e da computação ubíqua na telemedicina, propondo um sistema de saúde pessoal que é adaptável, sensível ao contexto e com capacidades de diagnóstico a partir de sensores médicos.

As principais contribuições dadas deste trabalho são as seguintes:

- Uma arquitetura para um sistema pessoal de telemedicina, ubíquo e sensível ao contexto, com capacidades de diagnóstico geral e completamente adaptável. Esta proposta, integra a gestão de todos os componentes de um sistema sensível ao contexto, numa solução que pode ainda ser estendida dada a sua flexibilidade. Não foi observada nenhuma solução existente, que contemplasse todas estas características, simultaneamente.
- Uma solução adaptável para resolver o problema de inferência num contexto multi-sensor. Esta, é mais completa que uma solução puramente *event-driven* e poupa mais recursos do que uma solução que analise continuamente os dados dos sensores.
- Uma linguagem integrável com *Web Services*, que permite representar um conjunto de instruções básico, para a formulação de processos de diagnóstico.

Avaliando se as metas iniciais foram atingidas com base nos resultados obtidos, concluiu-se que o problema delimitado na afirmação desta tese foi ultrapassado. Foram realizados, testes e casos de estudo que comprovam que é possível integrar conhecimento médico

num sistema ubíquo, remota e dinamicamente. Contudo, os casos de estudo e testes realizados não comprovam o funcionamento total da solução integrada, embora tenham funcionado nos testes dos vários aspectos do sistema em separado, nomeadamente no ensaio oficial do *CAALYX*.

## 6.2 Trabalho Futuro

### 6.2.1 Linguagem das Observações

O sistema desenvolvido, utiliza uma linguagem *XML* para definir os diagnósticos e valores limite. Esta linguagem, não é *standard* da indústria e não cumpre uma álgebra de eventos temporais completa, embora contenha algumas instruções que incorporam a dimensão temporal. Um dos aspectos a melhorar, seria evoluir esta linguagem para uma linguagem que fosse compatível com um *standard*

### 6.2.2 Interoperabilidade

Existem muitos *standards* de interoperabilidade para sistemas médicos, e torna-se fundamental para questões de exploração, que o sistema seja compatível com *standards*, utilizados na Indústria. Deste modo, é possível descentralizar as fontes de informação, tornando-as interoperáveis localmente, sem a necessidade de existência de uma fonte de dados centralizada, tal como aconteceu neste trabalho.

### 6.2.3 Implementar arquitectura de *Plugins*

Neste momento, a adaptabilidade reduz-se às regras de diagnóstico e de valores limite. Por outro lado, as áreas à volta da sensibilidade ao contexto aplicadas à saúde, evoluem rapidamente em termos de sensores e em termos de capacidade de inferência. Deste modo, a arquitectura do sistema deverá evoluir de modo a suportar a alteração dinâmica dos componentes da aplicação e de incluir novos componentes remotamente. Deste modo, seria possível evoluir a aplicação e manter os custos de desenvolvimento baixos. Existem várias hipóteses tecnológicas que potencialmente podem permitir esse tipo de funcionalidades, tais como a adaptação de arquitecturas de componentes e *middleware*, já existentes para sistemas com estas características.

# Referências

- [AJT05] Oliver Amft, Holger Junker, and Gerhard Troster. Detection of eating and drinking arm gestures using inertial body-worn sensors. In *ISWC '05: Proceedings of the Ninth IEEE International Symposium on Wearable Computers*, pages 160–163, Washington, DC, USA, 2005. IEEE Computer Society.
- [AKT07] Oliver Amft, Martin Kusserow, and Gerhard Troster. Automatic identification of temporal sequences in chewing sounds. In *BIBM '07: Proceedings of the 2007 IEEE International Conference on Bioinformatics and Biomedicine*, pages 194–201, Washington, DC, USA, 2007. IEEE Computer Society.
- [ALO<sup>+</sup>04] Oliver Amft, Michael Lauffer, Stijn Ossevoort, Fabrizio Macaluso, Paul Lukowicz, and Gerhard Troster. Design of the qbic wearable computing platform. In *ASAP '04: Proceedings of the Application-Specific Systems, Architectures and Processors, 15th IEEE International Conference*, pages 398–410, Washington, DC, USA, 2004. IEEE Computer Society.
- [AT08] Oliver Amft and Gerhard Tröster. Recognition of dietary activity events using on-body sensors. *Artif. Intell. Med.*, 42(2):121–136, 2008.
- [ATLS06] Oliver Amft, Gerhard Troster, Paul Lukowicz, and Corina Schuster. Sensing muscle activities with body-worn sensors. In *BSN '06: Proceedings of the International Workshop on Wearable and Implantable Body Sensor Networks*, pages 138–141, Washington, DC, USA, 2006. IEEE Computer Society.
- [BG03] M. Beigl and H. Gellersen. Smart-its: An embedded platform for smart objects, 2003. [citeseer.ist.psu.edu/beigl03smartits.html](http://citeseer.ist.psu.edu/beigl03smartits.html).
- [car] Carecompanion home monitoring system. <http://www.hometelehealthltd.co.uk>, acedido a ultima vez em 21 de Maio de 2008.
- [CDKF06] George Ganiatsas Christos D Katsis and Dimitrios I Fotiadis. An integrated telemedicine platform for the assessment of affective physiological states. *Diagnostic Pathology*, 2006.
- [CK00] Guanling Chen and David Kotz. A survey of context-aware mobile computing research. Technical report, Hanover, NH, USA, 2000.
- [CM01] Adam Cheyer and David Martin. The open agent architecture. *Autonomous Agents and Multi-Agent Systems*, 4(1-2):143–148, 2001.
- [cor] Corscience. <http://www.corscience.de/>, acedido a ultima vez em 23 de Julho de 2008.

## REFERÊNCIAS

- [Cor91] Daniel Corkill. Blackboard Systems. *AI Expert*, 6(9), January 1991. <http://mas.cs.umass.edu/paper/218>.
- [DCI] The disappearing computer initiative. <http://www.disappearing-computer.net/>, acessido a ultima vez em 20 de Julho de 2008.
- [Dey00] Anind Kumar Dey. *Providing architectural support for building context-aware applications*. PhD thesis, Atlanta, GA, USA, 2000. Director-Gregory D. Abowd.
- [dic] Dicoems project. [http://www.dicoems.com/site/index.php?option=com\\_frontpage&Itemid=1](http://www.dicoems.com/site/index.php?option=com_frontpage&Itemid=1), acessido a ultima vez em 22 de Junho de 2008.
- [ea] Mike Botts et al. Opengis sensor model language (sensorml) implementation specification. Technical report, Open Geospatial Consortium.
- [eme] The emergency monitoring and prevention. <http://www.emerge-project.eu/>, acessido a ultima vez em 23 de Junho de 2008.
- [FAB<sup>+</sup>02] G. Ferguson, J. Allen, N. Blaylock, D. Byron, N. Chambers, M. Dzikovska, L. Galescu, X. Shen, R. Swier, and M. Swift. The medication advisor project: Preliminary report. Technical report, Rochester, NY, USA, 2002.
- [GHJV95] Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides. *Design patterns: elements of reusable object-oriented software*. Addison-Wesley Professional, 1995.
- [GR05] Simson Garfinkel and Beth Rosenberg. *RFID: Applications, Security, and Privacy*. Addison-Wesley Professional, 2005.
- [GSSS02] David Garlan, Dan Siewiorek, Asim Smalagic, and Peter Steenkiste. Project aura: Towards distraction-free pervasive computing. *IEEE Pervasive Computing, special issue on Integrated Pervasive Computing Environments*, 1(2):22–31, April 2002.
- [heaa] Health buddy. <http://www/medicare.com.au>, acessido a ultima vez em 22 de Julho de 2008.
- [heab] The health service 24. [www.healthservice24.com/](http://www.healthservice24.com/), acessido a ultima vez em 19 de Maio de 2008.
- [int] A virtual reality intelligent multi-sensor wearable system for phobias' treatment. <http://www.intrepid-project.org/>, acessido a ultima vez em 19 de Maio de 2008.
- [JALT08] Holger Junker, Oliver Amft, Paul Lukowicz, and Gerhard Tröster. Gesture spotting with body-worn inertial sensors to detect user activities. *Pattern Recogn.*, 41(6):2010–2024, 2008.
- [Kja07] Kristian Ellebaek Kjaer. A survey of context-aware middleware. In *SE'07: Proceedings of the 25th conference on IASTED International Multi-Conference*, pages 148–155, Anaheim, CA, USA, 2007. ACTA Press.

## REFERÊNCIAS

- [lif] Lifeshirt by vivometrics. <http://www.vivometrics.com/research/index.php>, acessido a ultima vez em 15 de Maio de 2008.
- [LSD<sup>+</sup>06] J. Luprano, J. Sola, S. Dasen, J. M. Koller, and O. Chetelat. Combination of body sensor networks and on-body signal processing algorithms: the practical case of myheart project. In *BSN '06: Proceedings of the International Workshop on Wearable and Implantable Body Sensor Networks*, pages 76–79, Washington, DC, USA, 2006. IEEE Computer Society.
- [MGR<sup>+</sup>99] Nelson Minar, Matthew Gray, Oliver Roup, Raffi Krikorian, and Pattie Maes. Hive: Distributed agents for networking things. *asama*, 0:118, 1999.
- [Neu02] Christopher Neufelt. Amon: A wearable medical computer for high risk patients. In *ISWC '02: Proceedings of the 6th IEEE International Symposium on Wearable Computers*, page 133, Washington, DC, USA, 2002. IEEE Computer Society.
- [OGC] Open geospatial consortium. <http://www.opengeospatial.org>, acessido a ultima vez em 19 de Maio de 2008.
- [par] Palo alto research center. <http://www.parc.com/>, acessido a ultima vez em 28 de Julho de 2008.
- [per] Personal health monitoring system. <http://www.phmon.de/englisch/index.html>, acessido a ultima vez em 22 de Junho de 2008.
- [phi] Philips medical home page. <http://www.medical.philips.com/index.html>, acessido a ultima vez em 19 de Maio de 2008.
- [Phi99] B. Phillips. Metaglu: A programming language for multi agent systems, 1999.
- [Pro06] Florian Probst. Ontological analysis of observations and measurements. In *GIScience*, pages 304–320, 2006.
- [RABB07] Julian Randall, Oliver Amft, Jürgen Bohn, and Martin Burri. Luxtrace: indoor positioning using building illumination. *Personal Ubiquitous Comput.*, 11(6):417–428, 2007.
- [Ric] Katarzyna Wac Richard. Mobile health care over 3g networks: The mobihealth pilot system and service.
- [Sch95] Bill N. Schilit. *A System Architecture for Context-Aware Mobile Computing*. PhD thesis, 1995.
- [Sea01] M. Satyanarayanan and et al. Pervasive computing: Vision and challenges, 2001.
- [SN] Norbert Streitz and Paddy Nixon. The disappearing computer.
- [SrCL<sup>+</sup>05] Victor Shnayder, Bor rong Chen, Konrad Lorincz, Thaddeus R. F. Fulford Jones, and Matt Welsh. Sensor networks for medical care. In *SenSys '05: Proceedings of the 3rd international conference on Embedded networked sensor systems*, pages 314–314, New York, NY, USA, 2005. ACM.

## REFERÊNCIAS

- [Sta96] Thad Starner. Human-powered wearable computing. *IBM Systems Journal*, 35(3/4):618–629, 1996. [citeseer.ist.psu.edu/186175.html](http://citeseer.ist.psu.edu/186175.html).
- [tel] Telemedcare system. <http://www/medcare.com.au>, acessido a ultima vez em 22 de Julho de 2008.
- [TLL05] Binh An Truong, Young-Koo Lee, and Sung-Young Lee. Modeling and reasoning about uncertainty in context-aware systems. In *ICEBE '05: Proceedings of the IEEE International Conference on e-Business Engineering*, pages 102–109, Washington, DC, USA, 2005. IEEE Computer Society.
- [vit] Vital jacket. <http://www.vitaljacket.com/>, acessido a ultima vez em 25 de Junho de 2008.
- [WB97] Mark Weiser and John Seely Brown. The coming age of calm technology. pages 75–85, 1997.
- [wei] Mark weiser. <http://www.ubiq.com/hypertext/weiser/weiser.html>, acessido a ultima vez em 28 de Julho de 2008.
- [Wei98] Mark Weiser. The invisible interface: Increasing the power of the environment through calm technology. In *CoBuild '98: Proceedings of the First International Workshop on Cooperative Buildings, Integrating Information, Organization, and Architecture*, page 1, London, UK, 1998. Springer-Verlag.
- [Wei99] Mark Weiser. The computer for the 21st century. *SIGMOBILE Mob. Comput. Commun. Rev.*, 3(3):3–11, 1999.
- [YJ02] Joseph W. Yoder and Ralph E. Johnson. The adaptive object-model architectural style. In *WICSA 3: Proceedings of the IFIP 17th World Computer Congress - TC2 Stream / 3rd IEEE/IFIP Conference on Software Architecture*, pages 3–27, Deventer, The Netherlands, The Netherlands, 2002. Kluwer, B.V.