

Sistema Android para a Interligação de Dispositivos Médicos Usando Normas IEEE

Por

Patrick Cunha Vidal

Orientador: Doutor João Agostinho Batista de Lacerda Pavão

Co-orientador: Doutor Luís José Calçada Torres Pereira



Dissertação submetida à
UNIVERSIDADE DE TRÁS-OS-MONTES E ALTO DOURO
para obtenção do grau de
MESTRE

em Engenharia Electrotécnica e de Computadores, de acordo com o disposto no
DR – I série– N.º151, Decreto-Lei n.º 115/2013 de 7 de Agosto e no
Regulamento de Estudos Conducente ao Grau de Mestre da UTAD
DR, 2.ª série – N.º133 de 13 de Julho de 2016

Sistema Android para a Interligação de Dispositivos Médicos Usando Normas IEEE

Por

Patrick Cunha Vidal

Orientador: Doutor João Agostinho Batista de Lacerda Pavão

Co-orientador: Doutor Luís José Calçada Torres Pereira

Dissertação submetida à

UNIVERSIDADE DE TRÁS-OS-MONTES E ALTO DOURO

para obtenção do grau de

MESTRE

em Engenharia Electrotécnica e de Computadores, de acordo com o disposto no

DR – I série– Nº151, Decreto-Lei n.º 115/2013 de 7 de Agosto e no

Regulamento de Estudos Conducente ao Grau de Mestre da UTAD

DR, 2.ª série – Nº133 de 13 de Julho de 2016

Orientação Científica :

Doutor João Agostinho Batista de Lacerda Pavão

Professor Auxiliar do
Departamento de Engenharias
Escola de Ciências e Tecnologia da Universidade de Trás-os-Montes e Alto Douro

Doutor Luís José Calçada Torres Pereira

Professor Auxiliar do
Departamento de Engenharias
Escola de Ciências e Tecnologia da Universidade de Trás-os-Montes e Alto Douro

"You are the average of the five people you spend the most time with."

Jim Rohn

Sistema Android para a
Interligação de Dispositivos Médicos
Usando Normas IEEE

Patrick Cunha Vidal

Submetido na Universidade de Trás-os-Montes e Alto Douro
para o preenchimento dos requisitos parciais para obtenção do grau de
Mestre em Engenharia Electrotécnica e de Computadores

Resumo — Os dispositivos pessoais de saúde eletrónicos têm estado em constante desenvolvimento e têm sido beneficiados com a introdução das tecnologias sem fios de forma a poder enviar os dados de saúde para outros dispositivos de recolha de dados e que posteriormente sejam enviados para uma base de dados. O facto de existirem inúmeros dispositivos pessoais de saúde das mais variadas marcas e modelos que possuem tecnologias sem fios pode induzir a problemas de interoperabilidade que podem ser resolvidos usando protocolos de comunicação normalizados. O ISO/IEEE 11073 é uma norma em ascensão, própria para comunicação de dados de saúde e pode usar tecnologias sem fios, pelo que o objetivo global deste trabalho consiste em implementar uma aplicação Android que teste o *standard* ISO/IEEE 11073 PHD usando a biblioteca *open-source* Antidote.

Palavras Chave: Android, Antidote library, Bluetooth, Dispositivo pessoal de saúde, Interoperabilidade, ISO/IEEE 11073 PHD.

Development of a Electrical Conductivity Measurement System for Biological Tissues

Patrick Cunha Vidal

Submitted to the University of Trás-os-Montes and Alto Douro
in partial fulfillment of the requirements for the degree of
Master of Science in Electrical Engineering and Computers

Abstract — Electronic personal health devices have been in constant development and have benefited from the introduction of wireless technologies in order to be able to send health data to other data collection devices and then sent to a database. The fact that there are numerous personal health devices of the most varied brands and models that have wireless technologies can induce interoperability problems, which can be solved using standard communication protocols. ISO / IEEE 11073 is a growing standard for health data communication, which can use wireless technologies. The overall goals of this work is to implement an Android application that tests the ISO / IEEE 11073 PHD standard using the Antidote library.

Key Words: Android, Antidote library, Bluetooth, Personal health device, Interoperability, ISO / IEEE 11073 PHD.

Agradecimentos

O meu agradecimento aos meus orientadores, Doutor João Pavão e Doutor Luís Pereira por toda a ajuda, dedicação, preocupação e tempo disponibilizado para o desenvolvimento deste trabalho, pelas palavras motivadoras e sugestões nos momentos de maiores dificuldades e pela confiança depositada nas minhas escolhas. Agradeço também à Doutora Rute Bastardo pela grande ajuda no *design* da interface Android. Quero agradecer também aos colegas da Universidade Técnica de Viena pela disponibilização do dispositivo pessoal de saúde.

Quero agradecer aos meus pais por acreditarem sempre em mim, pelo apoio sempre presente e pela tomada de difíceis decisões de vida que possibilitaram a minha formação. Ao meu irmão pelas palavras não menos importantes mas originais que fazem parte do dia a dia, à minha restante família e à Marta Pires pelo grande apoio e também aos meus amigos por me acompanharem ao longo do percurso académico.

A todos, um sincero obrigado!

Índice geral

Resumo	ix
<i>Abstract</i>	xi
Agradecimentos	xiii
Índice de tabelas	xvii
Índice de figuras	xix
Glossário, acrónimos e abreviaturas	xxi
1 Introdução	1
1.1 Motivação e Objetivos	3
1.2 Organização da dissertação	4
2 Dispositivos Pessoais de Saúde e Interoperabilidade	7
2.1 Dispositivos Pessoais de Saúde	7
2.2 Tecnologias de Comunicações Sem Fios nos Dispositivos Pessoais de Saúde	9
2.3 Normas de Interoperabilidade e Organizações que as Promovem	10
2.3.1 Integrating the Healthcare Enterprise (IHE)	12
2.3.2 Personal Connected Health Alliance (PCHA)	13

3	Comunicação entre Dispositivos Pessoais de Saúde	17
3.1	ISO/IEEE 11073 Personal Health Device	17
3.1.1	Agentes, Gestores e Transporte	19
3.1.2	Visão Geral da família 11073 PHD	19
3.1.3	Domain Information Model	21
3.1.4	Service Model	23
3.1.5	Communication Model	24
3.1.6	ASN.1	31
3.2	Bluetooth	31
3.2.1	Funcionamento Técnico	32
3.2.2	Arquitetura	33
3.2.3	Versões Bluetooth	36
3.3	HDP	37
4	Interface Android para um Dispositivo Pessoal de Saúde	41
4.1	Materiais e Métodos	41
4.2	Antidote	43
4.2.1	Componentes do Antidote	43
4.3	Implementação usando o Antidote	45
4.3.1	Implementação de uma Interface do Utilizador	47
4.4	Resultados Obtidos	49
5	Conclusão e trabalho futuro	53
	Referências bibliográficas	55
A	Implementação detalhada	63

Índice de tabelas

3.1	Classe dos dispositivos Bluetooth	32
4.1	Características técnicas dos oxímetro Nonin 9560	46

Índice de figuras

1.1	Caso de uso	2
2.1	Dispositivos Pessoais de saúde	8
2.2	Arquitetura da PCHA	13
2.3	Interface da PCHA	14
3.1	<i>Stack</i> do ISO/IEEE 11073 PHD	20
3.2	Arquitetura do ISO/IEEE 11073 PHD	21
3.3	<i>Domain Information Model</i>	23
3.4	Máquina de Estados de um <i>Manager</i> 11073	26
3.5	Estado <i>Connected</i> e <i>Disconnected</i>	27
3.6	Estado <i>Unassociated</i>	28
3.7	Estado <i>Associating</i>	28
3.8	Estado <i>Associated</i>	29
3.9	Estado <i>Configuring</i>	29
3.10	Estado <i>Operating</i>	30
3.11	Estado <i>Disassociating</i>	30
3.12	Exemplo de uma <i>scatternet</i>	33
3.13	Arquitetura da <i>Stack</i> protocolar do Bluetooth	35

3.14 HDP inserido na <i>Stack</i> Bluetooth	39
4.1 Componentes da <i>stack</i> do Antidote	43
4.2 Esquema de montagem do sistema proposto	46
4.3 Excerto de código do XML	47
4.4 Menu inicial da aplicação elaborada	48
4.5 Menu do oxímetro da aplicação elaborada	48
4.6 Medição da percentagem de oxigénio presente no sangue e dos bati- mentos cardíacos por minuto	51

Glossário, acrónimos e abreviaturas

Lista de acrónimos

Sigla	Expansão
AARQ	<i>Association Request</i>
AARE	<i>Association Response</i>
ABRT	<i>Association Abort</i>
ACL	<i>Asynchronous Connection Less</i>
ANSI	<i>American National Standards Institute</i>
APDU	<i>Application Protocol Data Unit</i>
ASN.1	<i>Abstract Syntax Notation 1</i>
BER	<i>Basic Encoding Rules</i>
BSIG	<i>Bluetooth Special Interest Group</i>
CDG	<i>Continua Design Guidelines</i>
CEN	<i>The European Committee for Standardization</i>
CHA	<i>Continua Health Alliance</i>
DER	<i>Distinguished Encoding Rules</i>
DICOM	<i>Digital Imaging and Communication in Medicine</i>

Sigla	Expansão
DIM	<i>Domain Information Model</i>
EHR	<i>Electronic Health Record (Registo Eletrónico de Saúde)</i>
FHSS	<i>Spread Spectrum Frequency Hopping</i>
GAP	<i>Generic Access Profile</i>
HCI	<i>Host Controller Interface</i>
HDP	<i>Health Device Profile</i>
HIMSS	<i>Healthcare Information and Management System Society</i>
HIS	<i>Healthcare Information System</i>
HL7	<i>Health Level 7</i>
HS	<i>High Speed</i>
ID	<i>Identity</i>
IDE	<i>Integrated Development Environment</i>
IEEE	<i>Institute of Electrical and Electronics Engineers</i>
IHE	<i>Integrating the Healthcare Enterprise</i>
IP	<i>Internet Protocol</i>
IPQ	<i>Instituto Português da Qualidade</i>
ISM	<i>Industrial, Scientific and Medical</i>
ISO	<i>International Standards Organization</i>
ITU	<i>International Telecommunication Union</i>
JNI	<i>Java Native Interface</i>
L2CAP	<i>Logical Link Control and Adaptation</i>
LE	<i>Low Energy</i>
LMP	<i>Link Manager Protocol</i>
MCAP	<i>Multi-Channel Adaptation Protocol</i>
MDER	<i>Medical Device Encoding Rules</i>
MDS	<i>Medical Device System</i>
MTU	<i>Maximum Transfer Unit</i>

Sigla	Expansão
NDK	<i>Native Development Kit</i>
NFC	<i>Near Field Connection</i>
OBEX	<i>Object Exchange Protocol</i>
OMS	Organização Mundial da Saúde
PAN	<i>Personal Area Network</i>
PC	<i>Personal Computer</i>
PCHA	<i>Personal Connected Health Alliance</i>
PHD	<i>Personal Health Device</i>
PHG	<i>Personal Health Gateway</i>
PM-Store	<i>Persistent Metric Store</i>
PM-Segment	<i>Permanent Metric Segment</i>
PPP	<i>Point-to-Point Protocol</i>
PRST	<i>Presentation</i>
RFCOMM	<i>Radio Frequency Communications</i>
RLRE	<i>Association Release Response</i>
RLRQ	<i>Association Release Request</i>
ROER	<i>Remote Invoke Error</i>
ROIV	<i>Remote Invoke</i>
RORJ	<i>Remote Invoke Reject</i>
RORS	<i>Remote Invoke Response</i>
SAR	<i>Segmentation and Reassembly</i>
SCO	<i>Synchronous Connection Oriented</i>
SDP	<i>Service Discovery Protocol</i>
SGML	<i>Standard Generalized Markup Language</i>
SPP	<i>Serial Port Profile</i>
TC	<i>Technical Committees</i>
TCP	<i>Transmission Control Protocol</i>

Sigla	Expansão
TCS	<i>Telephony Control Protocol Specification</i>
UDP	<i>User Datagram Protocol</i>
UI	<i>User Interface</i>
USB	<i>Universal Serial Bus</i>
WAN	<i>Wide Area Network</i>
XER	<i>XML Encoding Rules</i>
XML	<i>eXtensible Markup Language</i>

Lista de abreviaturas

Abreviatura	Significado(s)
e.g.	por exemplo
et al.	e outros (autores)
etc.	etcetera, outros
i.e.	isto é, por conseguinte
vid.	veja-se, ver
vs.	versus, por comparação com
sec.	século

1

Introdução

Desde há muitos anos que a Humanidade tem vindo a alterar a sua qualidade de vida através do melhoramento dos cuidados de saúde. Para alguns autores, a qualidade de vida e a saúde são termos indissociáveis, contudo, a saúde não é o único fator que influencia a nossa qualidade de vida mas tem uma importância fulcral [1].

Segundo a Organização Mundial da Saúde (OMS), a saúde é “um estado de completo bem-estar físico, mental e social e não apenas a ausência de doença” [2]. Esta definição é bastante abrangente, envolvendo também aspetos socioeconómicos para além da saúde física e mental, o que faz com que o estado de completa saúde seja algo ideal, inatingível [3]. A saúde é um fator essencial da vida humana, capaz de providenciar a um indivíduo bem-estar, que lhe garanta um desempenho satisfatório em diversos planos: a nível físico, psicológico e social [4].

De forma a garantir uma boa qualidade de vida é necessário ter hábitos saudáveis, cuidar bem do corpo, ter uma alimentação equilibrada entre outros vários hábitos que façam um indivíduo sentir-se bem [5]. Por outro lado a saúde pode ser afetada de forma negativa, ou seja, a ocorrência de uma doença debilitante que constitui, ou não risco de vida, o declínio natural da saúde de uma pessoa idosa, declínio mental, etc. Todas estas situações são castradoras da qualidade de vida das pessoas [1].

Com o avanço das tecnologias e da divulgação das mesmas surgiram os dispositivos de monitorização de estado de saúde, que permitem aos profissionais de saúde prevenir, diagnosticar, monitorizar e tratar doenças, ou até permitir à população obter um maior controlo sobre o seu estado de saúde ao utilizarem os dispositivos pessoais de saúde.

Nos dias de hoje é fácil aceder a dispositivos pessoais de saúde, sendo que estes estão disponíveis no mercado e muitos deles estão equipados de tecnologias *wireless*, o que, combinado com os *smartphones* torna possível às pessoas o controlo individual da saúde.

Na figura 1.1 está ilustrado um exemplo de um caso de uso destes dispositivos.

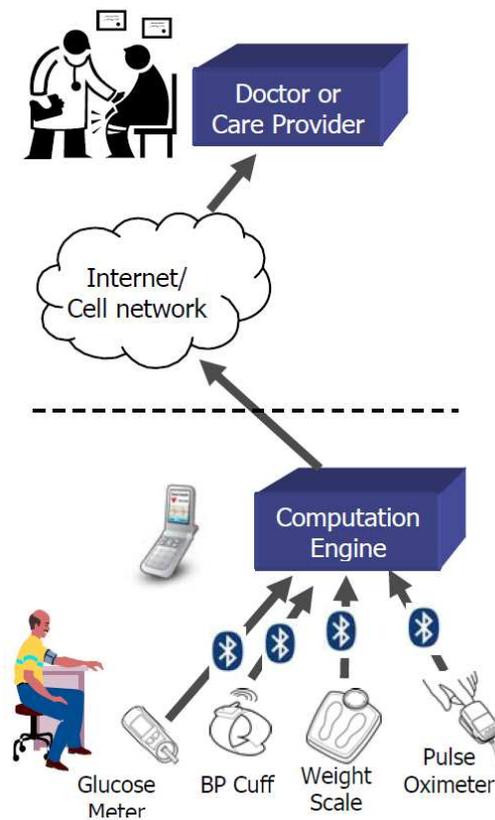


Figura 1.1 – Caso de uso [6].

Neste caso de uso está um paciente que possui em casa diversos dispositivos médicos que fazem medições e enviam os dados para um sistema de recolha de dados (e.g.

smartphone ou PC). Estes dados são enviados dos sistemas de recolha de dados para uma base de dados através de uma rede segura (e.g. através da internet) para mais tarde um profissional de saúde ter acesso e assim avaliar a evolução da saúde do paciente.

A visualização dos dados recolhidos pelos dispositivos é possível usando aplicações próprias do dispositivo pessoal de saúde, o que poderá ser um problema quando são utilizados diversos dispositivos de diferentes fabricantes, pois leva a uma sobrecarga de aplicações específicas para cada um deles, assim como protocolos proprietários das várias marcas e modelos. Geralmente não é possível a troca de dados entre estas diferentes aplicações com vista, por exemplo, a um registo eletrónico de saúde único correspondente a um indivíduo, que possa ser enviado de forma integrada ao fornecedor de cuidados de saúde ou ao médico. Esta falta de capacidade de trocar dados entre as diferentes aplicações traduz-se num problema de interoperabilidade que pode ser resolvido usando um *standard*, neste caso o *standard* ISO/IEEE 11073.

Neste projeto pretende-se desenvolver e testar a norma IEEE 11073 usando a *stack* do Antidote desenvolvido pela Signove [7], a funcionar em vários dispositivos médicos, coordenados por um dispositivo Android a funcionar como gestor.

1.1 Motivação e Objetivos

O IEEE tem funções de grande importância como a criação, aprovação e divulgação de normas técnicas em diversos setores incluindo a comunicação da informação na saúde informática. De forma a querer manifestar o interesse na normalização das mensagens dos dispositivos pessoais de saúde para a comunicação com outros sistemas surge o interesse na divulgação e implementação da família de normas ISO/IEEE 11073 aliado à explosão de dispositivos portáteis de medição de sinais relacionados com a saúde.

A contribuição para a divulgação da família de normas ISO/IEEE, a contribuição

para a conceção de um equipamento de gestão de dispositivos ligados em modo concorrential, a sua generalização nos dispositivos móveis e a existência de grupos de trabalho a desenvolver projetos similares e em parceria com a University of Applied Sciences Technikum Wien justifica a motivação deste estudo.

Com este estudo é pretendido implementar a norma em estudo numa aplicação, não expondo os detalhes da comunicação com o dispositivos pessoal de saúde na interface do utilizador. Para a concretização deste estudo, foram estabelecidos diversos objetivos que tornarão possível a implementação da norma em estudo:

1. Estudo da interoperabilidade de dispositivos pessoais de saúde;
2. Levantamento de *software open-source* que permita a implementação da norma IEEE 11073;
3. Implementação de uma aplicação Android para interface com dispositivos pessoais de saúde.

1.2 Organização da dissertação

Esta dissertação encontra-se estruturada em cinco capítulos. No presente Capítulo fez-se um enquadramento do tema dos dispositivos pessoais de saúde e da importância das questões da interoperabilidade e apresentou-se a motivação e objetivos do trabalho.

O Capítulo 2 inicia-se com uma abordagem aos dispositivos pessoais de saúde, onde se enumeram alguns exemplos desses dispositivos e tecnologias sem fios para a comunicação. Seguidamente faz-se um levantamento das organizações responsáveis pela criação de normas e das entidades que promovem o seu uso.

A norma IEEE 11073, em estudo, e a tecnologia usada para a comunicação com o respetivo perfil de saúde serão apresentadas no Capítulo 3.

A biblioteca utilizada e o procedimento para a implementação assim como os resultados encontram-se no Capítulo 4.

A finalizar a dissertação, no Capítulo 5, faz-se uma discussão dos resultados e apresentam-se algumas perspectivas de trabalho futuro.

2

Dispositivos Pessoais de Saúde e Interoperabilidade

Este capítulo inicia-se com uma apresentação geral dos dispositivos pessoais de saúde a que se segue a apresentação de tecnologias para a comunicação dos dados na área da saúde. É introduzido o conceito de interoperabilidade e são apresentadas algumas das maiores organizações que desenvolvem ou promovem normas relacionadas com esta área.

2.1 Dispositivos Pessoais de Saúde

Os dispositivos pessoais de saúde são importantes instrumentos de saúde que têm benefícios terapêuticos e são usados para medir ou monitorizar funções do corpo. São frequentemente utilizados quer por profissionais de saúde quer por leigos, e o impacto deles na saúde e nas despesas com cuidados de saúde são cada vez mais significativos. Muitas vezes os dispositivos médicos eletrónicos são confundidos com dispositivos eletrónicos que monitorizam dados fisiológicos também conhecidos como dispositivos pessoais de saúde. Existe uma barreira entre estes dois tipos de dispositivos que possuem um objetivo comum. Um dispositivo eletrónico que monitoriza dados fisiológicos coberto por várias diretivas da Comunidade Europeia, certificado

pelo Infarmed e utilizado por profissionais de saúde é considerado dispositivo médico, no entanto o mesmo dispositivo utilizado fora do contexto clínico, em que os resultados não são utilizados por um profissional de saúde, ajudando apenas o utilizador a monitorizar e conhecer os seus dados fisiológicos, somente é considerado dispositivo pessoal de saúde. Tanto os dispositivos médicos como os dispositivos pessoais de saúde são destinados a serem utilizados para fins comuns aos medicamentos tais como prevenir, diagnosticar ou até tratar uma doença mas não podem atingir o seu fim através de ações farmacológicas, metabólicas ou imunológicas, pois isso é a função dos medicamentos e assim entende-se a diferença entre dispositivos médicos e medicamentos [8, 9]. Na Figura 2.1 estão ilustrados alguns dispositivos pessoais de saúde.



Figura 2.1 – Dispositivos Pessoais de Saúde [10–13].

2.2 Tecnologias de Comunicações Sem Fios nos Dispositivos Pessoais de Saúde

Muito dos dispositivos pessoais de saúde disponíveis no mercado são capazes de enviar informações remotamente para uma estação base através de tecnologias *wireless*. Dispositivos como balanças, termómetros ou oxímetros, mencionados na Figura 2.1, usam tecnologias como o Bluetooth ou ZigBee para enviar os resultados para um computador, *tablet* ou *smartphone*. É vantajoso não só para o paciente mas também para toda a comunidade médica. As vantagens incluem [14, 15]:

- Eliminação da necessidade de introdução manual dos dados no registo clínico do paciente;
- Atualizações dos dados mais rápidos e eficientes;
- Menor erro humano;
- Maior eficiência de trabalho;
- Menores custos de manutenção.

Uma estação base pode usar os dados provenientes desses diferentes dispositivos para melhor monitorizar a saúde dos pacientes mas a implementação de uma estação base que seja capaz de ler os dados provenientes de vários dispositivos diferentes pode ser problemática se cada dispositivo utilizar protocolos de comunicação proprietários ou os dados seguirem diferentes formatos de informação. Existe então um problema de interoperabilidade que faz com que seja complicado para os programadores obterem os dados numa estrutura normalizada independentemente do tipo de dispositivo, marca e modelo para depois poderem ser inseridos no registo clínico de saúde dos pacientes, o *Electronic Health Records* (EHRs) [15].

2.3 Normas de Interoperabilidade e Organizações que as Promovem

Na área da informática médica pretende-se que sistemas independentes tenham compatibilidade entre si, ou seja, que exista interoperabilidade. A interoperabilidade é definida como a “capacidade de comunicar, de executar programas ou de transferir dados entre várias unidades funcionais, graças à utilização de linguagens e de protocolos comuns exigindo poucos ou mesmo nenhuns conhecimentos do utilizador sobre as características específicas dessas unidades” [16]. De um modo geral, a interoperabilidade é uma característica de um produto ou sistema em que as interfaces são completamente compreendidas para trabalhar com outros produtos ou sistemas, tanto na implementação como no acesso, sem restrições [17].

Uma das formas de se conseguir interoperabilidade é com a implementação de normas para a troca de informação ao nível semântico e funcional entre sistemas heterogéneos, isto é, os sistemas terem a capacidade de compreender a informação trocada entre eles ao nível dos conceitos formais no domínio clínico.

O principal objetivo da normalização é a adoção de normas com o intuito de estabelecer soluções tecnológicas e económicas, com o consenso das partes interessadas, relativamente a assuntos com carácter repetitivo [18]. Por sua vez as normas são documentos resultantes de um consenso, aprovados por um organismo de normalização reconhecido, como por exemplo a *International Standards Organization* (ISO), que estabelece regras, guias ou características de produtos ou serviços [19].

As normas, também denominadas de standards, dividem-se em dois grupos, em standards de jure e em standards de facto. Os standards de jure também chamados de standards de acordo com a lei, são defendidos por uma organização formal de padrões e essa mesma organização ratifica cada norma desenvolvida através dos seus procedimentos oficiais e insere na norma o seu selo de aprovação. Os standards de facto são também denominados por padrões da atualidade, são amplamente adotados por uma indústria e o seus clientes. Estes padrões surgem quando uma grande massa

de indústrias gosta deles o suficiente para serem usados coletivamente e podem se tornar padrões de jure se forem aprovados por uma organização formal de standards [20].

Existem várias entidades responsáveis pelo desenvolvimento de normas de jure e algumas das que se destacam na informática da saúde são:

- *International Organization for Standardization (ISO)* - Criada em 1947 com sede em Genebra, é uma organização não governamental que tem como missão promover o desenvolvimento mundial de normas de forma a facilitar a troca internacional de serviços. É composta por membros de 163 organismos nacionais de normalização e através destes membros reúne especialistas para compartilhar conhecimentos e desenvolver normas internacionais seguindo princípios de voluntariedade, de consenso e de alcance. Em Portugal a ISO é representada pelo Instituto Português da Qualidade (IPQ) [21];
- *The European Committee for Standardization (CEN)* - O comité europeu de normalização é uma associação que reúne os organismos nacionais de normalização de 34 países europeus. Promove a normalização em relação a diversos setores e campos nomeadamente ar e espaço, defesa e segurança, saúde e segurança, saúde, entre outros. A CEN também apoia a normalização internacional cooperando com a ISO a fim de conseguir o objetivo de um standard aceite em todo o mundo [22, 23];
- *American National Standards Institute (ANSI)* - Fundada em 1918, é uma organização sem fins lucrativos que supervisiona a criação, a promulgação e o uso de milhares de normas e diretrizes em praticamente todos os setores no Estados Unidos da América. O ANSI promove o uso de standards estadunidenses a nível internacional. Como membro fundador da ISO, este tem acesso aos processos de desenvolvimento dos standards ISO [24, 25].

Apesar da importância da interoperabilidade no setor da saúde, existem desafios a ser enfrentados para a sua implementação. O desenvolvimento de *standards* pode

ser lento e a sua adoção ainda mais já que, quando se fala em saúde, fala-se num âmbito muito complexo, o que também torna a representação dos dados complexa, dificultando a adesão dos standards. Outro desafio e condição para a adoção dos *standards* é garantir a segurança e confidencialidade dos dados do paciente, caso contrário a confiança dos pacientes em relação à fiabilidade das aplicações pode estar em causa. Em contrapartida esta adoção simplifica a integração dos dados nos sistemas, reduzindo os custos de investimento e melhora a qualidade da prestação dos cuidados de saúde [26].

Dos standards de facto mais conhecidos e relacionados com a informática da saúde temos os que são descritos nos subcapítulos 2.3.1 e 2.3.2

2.3.1 Integrating the Healthcare Enterprise (IHE)

A IHE é uma entidade que resulta da colaboração dos profissionais de saúde e da indústria de tecnologia de informação para melhorar os sistemas informáticos no setor da saúde, compartilhando informação de uma forma consistente. “A iniciativa IHE é não só um processo mas também um fórum que incentiva a integração: define um fluxo de trabalho técnico para a implementação de normas de comunicação de modo a alcançar objetivos clínicos específicos” [27–29]. O objetivo da iniciativa IHE é apoiar a utilização das normas existentes, de entre elas existem duas normas relacionadas com a interoperabilidade, de grande importância na área da saúde.

A primeira é conhecida por *Digital Imaging and Communication in Medicine* (DICOM), uma norma utilizada na imagiologia médica que define um formato de imagem médica bem como um conjunto de mecanismos para a sua transferência, armazenamento e visualização [30].

A segunda é conhecida como *Health Level 7* (HL7), uma organização de desenvolvimento de standards sem fins lucrativos acreditada pela ANSI como criador oficial de *standards* para troca, integração, partilha e recuperação de informação na área médica. O HL7 estabelece o formato de mensagem que é comum a todo o tipo de informação como resultados das análises clínicas, registos de pacientes ou até as

faturas de modo a permitir a interoperabilidade e a partilha de informações clínicas, facilitando assim a integração da informação nos sistemas de informação [31, 32].

2.3.2 Personal Connected Health Alliance (PCHA)

A PCHA é uma organização que promove a interoperabilidade na saúde, nomeadamente na área dos dispositivos pessoais de saúde é a *Personal Connected Health Alliance* (PCHA). Fundada em 2006 como *Continua Health Alliance* (CHA), e agora associada à *mHealth Summit* (agora *Connected Health Conference*) e à *Healthcare Information and Management System Society* (HIMSS) desde abril 2014, é uma organização sem fins lucrativos que conta com mais de 150 empresas membro, incluindo a *Fujitsu*, *Intel*, *Nonin*, *Orange*, *Oracle* e *Phillips*. O objetivo da PCHA é construir um sistema de dispositivos de saúde interoperacionais, uma vez que a falta de interoperabilidade é um problema que afeta os cuidados de saúde, e permitir que a informação percorra desde a sua medição até ao registo eletrónico de saúde usando as diferentes interfaces e protocolos de comunicação (Figura 2.2) [33].

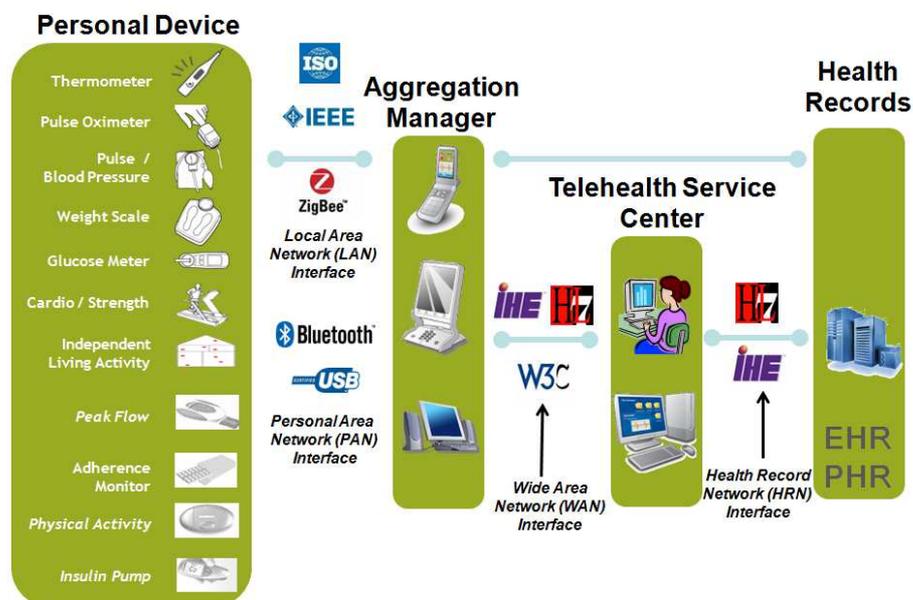


Figura 2.2 – Arquitetura da PCHA [34].

A arquitetura da PCHA é composta por três interfaces, entre elas, a *Personal Health Device Interface* (PHDI), *Service Interface* e *Healthcare Information System* (HIS) *Interface* (Figura 2.3).



Figura 2.3 – Interface da PCHA [35].

A PHDI é a interface situada entre o PHD (e.g. oxímetro, termómetro, balança, etc..) e o *Personal Health Gateway* (PHG) também conhecidos como *Agent* e *Manager* respetivamente, no âmbito da norma IEEE 11073. Nesta etapa os PHD e os PHG formam uma *Personal Area Network* (PAN) entre eles pois são dispositivos que necessitam de estar próximos uma vez que os PHD são utilizados no corpo. Protocolos como *Near Field Connection* (NFC), *Universal Serial Bus* (USB), *ZigBee* e *Bluetooth* serão os responsáveis para implementar as comunicações a curta distância.

A *Service Interface* situa-se entre o PHG e os serviços de saúde *back end* (e.g. *telehealth service center*) segundo normas ISO/IEEE 11073, tendo como papel principal, estabelecer uma interface interoperável para troca de mensagens entre eles dentro de uma *Wide Area Network* (WAN), uma vez que os PHGs não necessitam de estar perto do corpo, podendo estar distantes depois de obterem a medição e assim poder comunicar a vários metros de distância com os serviços de saúde *back end*. Nestas interfaces são usadas normas HL7 e especificações IHE para garantir a interoperabilidade.

A *Health Information System (HIS) Interface* permite a troca eletrónica de registos de saúde com outros serviços de saúde, mantendo a segurança através da confidencialidade, integridade e autenticação. Esta interface enquadra-se entre serviços de saúde *back end* e os EHRs [35, 36].

Para se conseguir a interoperabilidade na PCHA é necessário que um produto seja certificado, passando a usar o logotipo “Continua certified” e só então é garantido que este seja interoperável com outro produto também certificado, uma vez que ambos aderem aos *standards* definidos pela *Continua Design Guidelines (CDG)*. As CDG são reconhecidas como um *standard* internacional para sistemas de saúde pessoais através da *International Telecommunication Union (ITU)* e definem uma *framework* subjacente a vários *standards* e critérios para garantir a interoperabilidade entre dispositivos pessoais de saúde e respetivos dados [37, 38]. Estas diretrizes também compreendem uma arquitetura de referência e de ferramentas de certificação e serviços que são desenvolvidos com base em diversos *standards* tais como a família de protocolos ISO/IEEE 11073 Personal Health Devices (PHD) [39].

3

Comunicação entre Dispositivos Pessoais de Saúde

O presente capítulo foca-se na comunicação entre os diferentes dispositivos usados para a recolha e medição dos dados de saúde, realçando especificamente a norma ISO/IEEE 11073 assim como a tecnologia *wireless* escolhida para o presente trabalho, o Bluetooth.

3.1 ISO/IEEE 11073 Personal Health Device

As normas IEEE 11073 foram desenvolvidos com uma grande participação internacional, são e continuam a ser adotados como *standards* ISO através da ISO TC 215 *Health Informatics* e como normas europeias através do CEN TC 251 *Health Informatics*, mais especificamente como CEN ISO/IEEE 11073. O resultado final é um conjunto de normas internacionalmente harmonizadas que foram desenvolvidas e adotadas pelos países membros da ISO e do CEN [40].

A família de normas foi desenvolvida para funcionar em aparelhos médicos em ambiente hospitalar, também designados por *bed side monitors*. Os *bed side monitors* são aparelhos médicos que estão posicionados ao lado do paciente, ligados à corrente elétrica, com grande capacidade de processamento e funções de monitorização

de dados fisiológicos do paciente [41].

Com a evolução tecnológica surgiram dispositivos pessoais de saúde, capazes de medirem dados fisiológicos. Estes aparelhos têm como características serem alimentados a baterias, o que os limita na capacidade de processamento, e ao utilizarem uma comunicação sem fios leva-os a consumir ainda mais energia, o que tornou difícil a implementação do CEN ISO/IEEE 11073 nesses dispositivos [42]. De forma a resolver este problema, a norma foi ajustada e surgiu a família de normas ISO/IEEE 11073 PHD (Personal Health Device).

A família de normas ISO/IEEE 11073 - PHD visa proporcionar a interoperabilidade “plug-and-play” em tempo real, isto é, o utilizador só tem a estabelecer a ligação entre os dispositivos e o sistema automaticamente deteta, configura e comunica sem intervenção humana. Compreende-se que o conceito de tempo real admite a fração de segundos que determinada informação leva a ser tratada, processada e apresentada ao utilizador.

O ISO/IEEE 11073 - PHD baseia-se numa arquitetura composta pela norma base, o IEEE 11073-20601 *Application Profile - Optimized Exchange protocol*, e numa série de especializações normalizadas de dispositivos 11073-104xx, dos quais [43]:

- IEEE 11073-10404 - *Device Specialization* - especifica o modelo de dados utilizado num oxímetro;
- IEEE 11073-10407 - *Device Specialization* - especifica o modelo de dados utilizado num monitor de pressão sanguínea;
- IEEE 11073-10408 - *Device Specialization* - especifica o modelo de dados utilizado num termómetro
- IEEE 11073-10415 - *Device Specialization* - especifica o modelo de dados utilizado numa balança.

Para além destas existem muitas mais para os diversos tipos de dispositivos pessoais de saúde.

3.1.1 Agentes, Gestores e Transporte

No âmbito do IEEE 11073 existem os conceitos de *Agents* e *Managers* (Agentes e Gestores). No contexto desta dissertação um *Agent* trata-se do dispositivo pessoal de saúde, com limitação em termos de bateria, de interface com o utilizador e com pouco poder de processamento. É o gerador de informação clínica (e.g. oxímetro) e o responsável por encapsular a medição segundo a família de normas ISO/IEEE 11073. Em relação ao *Manager*, trata-se de um dispositivo com maior poder computacional, com poucas ou nenhuma limitações em termos de bateria, e é quem recolhe a informação clínica (e.g. *tablet*). O *Manager* também pode ter como função armazenar os dados para mais tarde reencaminhá-los para um serviço de saúde.

As tecnologias usadas para o transporte, isto é, as tecnologias para a transmissão da informação entre *Agents* e *Managers*, estão divididas em dois grupos: comunicações com fios (*wired*) e comunicações sem fios (*wireless*). Estas tecnologias podem ainda ser subdivididas em mais grupos dependendo do alcance da ligação. Em distâncias curtas normalmente são usadas tecnologias *Cabled PAN bus*, *Wireless PAN (Personal Area Network)*, *radio frequency ID*. Em distâncias maiores são usadas tecnologias como TCP/IP através de *Wireless LAN (Local Area Network)*. A comunicação entre *Agents* e *Managers* é assumida como uma ligação ponto-a-ponto, isto é, não existe um ponto central de controlo ou administração na rede, permitindo a partilha de dados. As comunicações são bidirecionais de forma a permitir segurança nas transações sendo que um *Agent* comunica apenas com um *Manager* num determinado momento, no entanto um *Manager* pode comunicar com mais do que um *Agent* em simultâneo usando conexões diferentes. Geralmente o *Agent* é quem inicia a conexão pois é este que sabe quando tem nova informação disponível [15, 42].

3.1.2 Visão Geral da família 11073 PHD

A família de normas ISO/IEEE 11073 PHD define uma *stack* de protocolos dividida em três níveis (Figura 3.1), em que a norma é independente dos canais de transporte, que se encontram na base da Figura 3.1, o que permite que seja feita uma escolha

entre as diferentes tecnologias de transportes existentes na CDG tais como USB, Bluetooth e *ZigBee*.

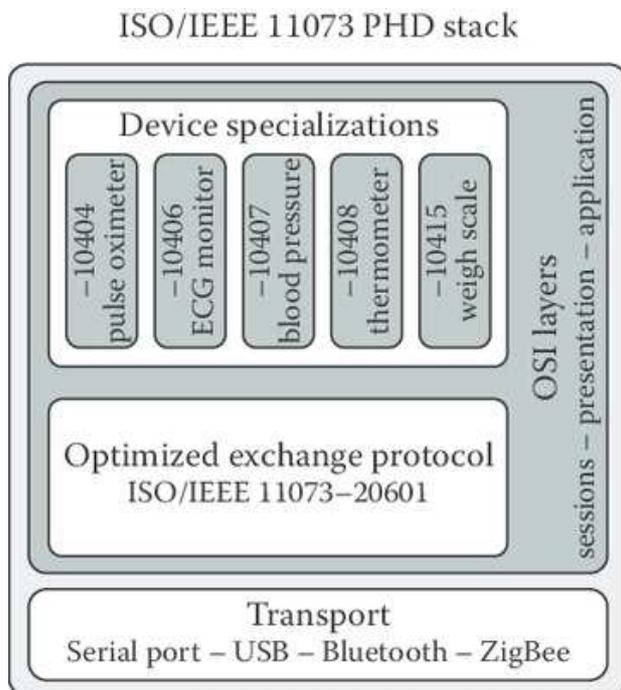


Figura 3.1 – Stack do ISO/IEEE 11073 PHD [44].

Acima da camada de transporte está a camada 11073-20601 *Optimized Exchange Protocol*. Esta camada descreve a máquina de estados, a forma como se gere a conexão, define comandos, eventos, formato dos dados e fornece as bases comuns para diferentes especializações dos dispositivos pessoais de saúde. Esta camada central define a arquitetura do *standard* (DIM, *Service Model* e *Communication Model*).

A Nomenclatura definida no ISO/IEEE 11073 usa um conjunto de termos, bem como os códigos numéricos correspondentes a esses termos, permitindo um idioma comum entre os dispositivos de medição de dados e os dispositivos de recolha de dados.

Cada *Agent* possui uma especialização diferente que define as capacidades do mesmo e a sua implementação de acordo com o *standard* 11073-20601. Cada especialização

descreve detalhes específicos para um tipo de *Agent* tais como atributos obrigatórios e o que estes significam. Descreve também configurações *standard* que este tipo de *Agent* pode adotar, por exemplo, na especialização 11073-10404, alusivo ao oxímetro, são obrigatórios dois atributos numéricos, um para as unidades em percentagem do oxigénio presente no sangue e outro para o pulso em batimentos por minuto.

Além desta norma se dividir em três níveis, também possui uma arquitetura de protocolos dividida em três modelos diferentes (Figura 3.2).

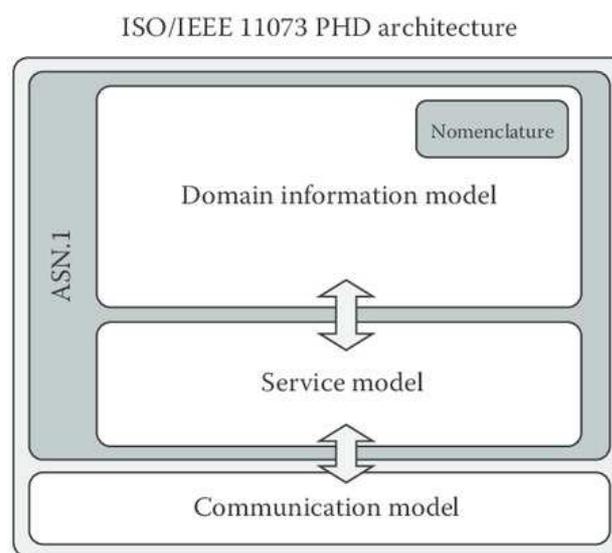


Figura 3.2 – Arquitetura do ISO/IEEE 11073 PHD [44].

3.1.3 Domain Information Model

O *Domain Information Model* (DIM) representa o dispositivo pessoal de saúde (*Agent*) e as suas funcionalidades através de um modelo hierárquico de estruturas que contêm um conjunto de objetos e atributos acerca das capacidades e configurações, assim como métodos que o *Manager* pode invocar. No sentido da programação orientada a objetos e de acordo com o *standard* IEEE 11073, um *Agent* é um conjunto de objetos, e cada classe possui um conjunto de atributos permissíveis mas, dependendo da classe, alguns atributos são obrigatórios, outros opcionais e outros proibidos [43, 45].

Cada *Agent* contém um único objeto *Medical Device System* (MDS), compondo o cimo da hierarquia de objetos. Por sua vez, cada MDS identifica o *Agent*, relata o seu estado e os seus atributos identificam-no para o *Manager*. Este objeto (MDS) dispõe de uma série de atributos, como por exemplo, o ID do sistema, informação de fabrico e especialização [15, 45].

A Figura 3.3 mostra a hierarquia DIM e as várias classes instanciadas como objetos que podem existir, particularmente [43]:

- *Metric*: A classe *Metric* é a classe base para todos os objetos que representam medições, estados e dados de contexto. Esta classe nunca é instanciada e serve como classe base para a *Numeric*, *Real Time Sample Array* e *Enumeration*;
- *Numeric*: representa uma única medida podendo ser um valor individual ou uma lista de números;
- *Real Time Sample Array*: representa as amostras contínuas ou formas de onda;
- *Enumeration*: representa informações de estado (códigos) ou anotações (texto);
- *PM-Store*: representa grandes quantidades de dados que foram adquiridos por um *Agent*. Fornece armazenamento a longo prazo e é composto por uma lista de *PM-Segments*. Funciona como um diretório num disco rígido;
- *PM-Segment*: um elemento da lista *PM-Store*, onde cada segmento guarda as medidas tiradas no passado. Os segmentos são enviados ao *Manager* quando pedidos. Funciona como um ficheiro num diretório;
- *Scanner*: os objetos do *Scanner* podem observar as medidas que estão a ser feitas no *Agent* e gerar eventos para informar o *Manager*. Os eventos podem ser relatórios regulares ou relatórios desencadeados por leituras anormais que são alarmadas. A classe *Scanner* nunca é instanciada, á semelhança da classe *Metric*. Esta é usada como classe base para a classe *Configurable Scanner*. Esta última é classe base para duas classes instanciáveis: *Episodic Scanner Configurable* e *Periodic Scanner Configurable*, reportam dados/eventos que

são separados por intervalos de tempo desconhecidos ou conhecidos, respectivamente.

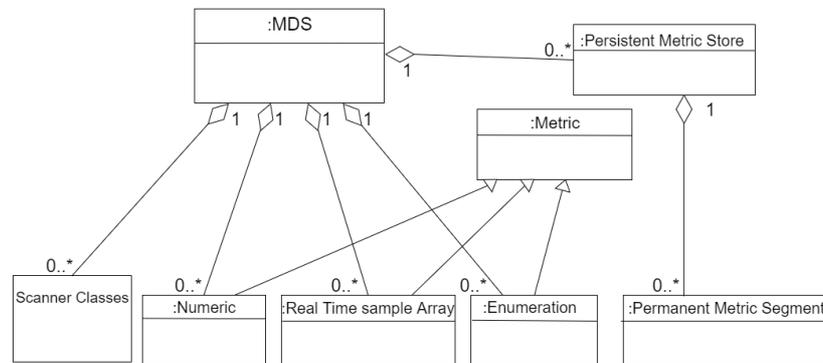


Figura 3.3 – Domain Information Model [43].

3.1.4 Service Model

O *Service Model* define serviços para a troca de dados. Estes serviços são mapeados para mensagens que são trocadas entre o *Agent* e o *Manager*. Também são usados para relatórios de dados e por funções de acesso fornecidas pelo *Agent*. Os serviços disponíveis são de associação e de acesso ao objeto [42, 46].

Os serviços de associação são responsáveis por gerir o processo de associação entre um *Agent* e um *Manager*. As mensagens deste serviço são:

- *Association request* (aaraq): O *Agent* inicia uma associação com o *Manager*.
- *Association response* (aare): O *Manager* responde ao pedido de associação.
- *Release request* (rlrq): O *Agent* ou *Manager* cancelam a associação.
- *Release response* (rlre): O *Agent* ou *Manager* respondem ao pedido de desassociação.
- *Abort* (abrt): O *Agent* ou *Manager* abortam a associação.

Os serviços de acesso ao objeto são responsáveis por aceder à informação dos objetos definidos no DIM, usados para relatórios de dados e por funções de acesso fornecidas pelo *Agent*. Os serviços de acesso são:

- *GET Service*: usado pelo *Manager* para recuperar valores de atributos de objetos. Disponível para objetos MDS e *PM-Store*;
- *SET Service*: usado pelo *Manager* para definir valores de atributo do objeto ao *Agent*. Os objetos *Scanner* suportam serviços "Set". O atributo de estado operacional do *Scanner* pode ser configurado para *enabled/disabled* fornecendo assim ao *Manager* controlo de fluxo de dados gerado pelo *Scanner*;
- *EVENT REPORT Service*: usado pelo *Agent* para enviar dados de medição ao *Manager*. Disponível para o objeto MDS, objetos *Scanner* e objetos *PM-Store*. Para o objeto MDS este serviço também é usado para enviar as informações de configuração do *Agent* para o *Manager*;
- *ACTION Service*: usado pelo *Manager* para invocar ações suportadas pelo *Agent*. Estes serviços são suportados pelos objetos MDS e objetos *PM-Store*.

3.1.5 Communication Model

O *Communication Model* descreve as características da comunicação, a máquina de estados da conexão e as interações válidas entre o *Agent* e o *Manager* em cada estado. Descreve também como as mensagens correspondentes a cada serviço são codificadas e decodificadas de e para o formato binário para serem transmitidos pelos canais de transporte.

Através de conexões ponto-a-ponto, um *Agent* comunica com um *Manager* e um *Manager* pode comunicar com vários *Agents*. Um *Application Protocol Data Unit* (APDU) deve ser processado atómicamente, e pode ser segmentado e depois remontado. O tamanho máximo de um APDU enviado a um *Manager* está limitado ao *Maximum Transmission Unit* (MTU) da especialização de cada dispositivo pessoal

de saúde (*Agent*) e a um valor máximo de 63Kb, enquanto que um APDU enviado a um *Agent* é de 8Kb. Importa salientar que a informação pode ser transmitida usando dois tipos de canais [42, 47]:

- canal confiável (*reliable channel*): Os APDUs são entregues ordenadamente, sem erros detetáveis, não duplicados e sem perdas. É possível que ocorram atrasos na transmissão por existir mais que uma tentativa de envio. A camada de comunicação deve indicar o início da conexão, a existência de desconexões, a falha no envio de APDUs e fazer o controlo de fluxo.
- canal de melhor esforço (*best-effort channel*): Os APDUs podem ser entregues de forma desordenada, duplicados, podendo mesmo chegar a uma taxa que cause a exaustão do *buffer* de receção. Os APDUS devem ter uma sequência numérica ao nível da aplicação.

Os estados que um *Manager* pode assumir são [48]:

- Disconnected
- Connected
 - * Associating
 - * Associated
 - Waiting for Configuration
 - Checking Configuration
 - Operating
 - * Disassociating
 - * Unassociated

Alguns estados são dependentes de outros, por exemplo, é necessário que o dispositivo esteja associado com o *Manager* para estar em modo de operação. A Figura 3.4 ilustra com detalhe a máquina de estados para um *Manager*.

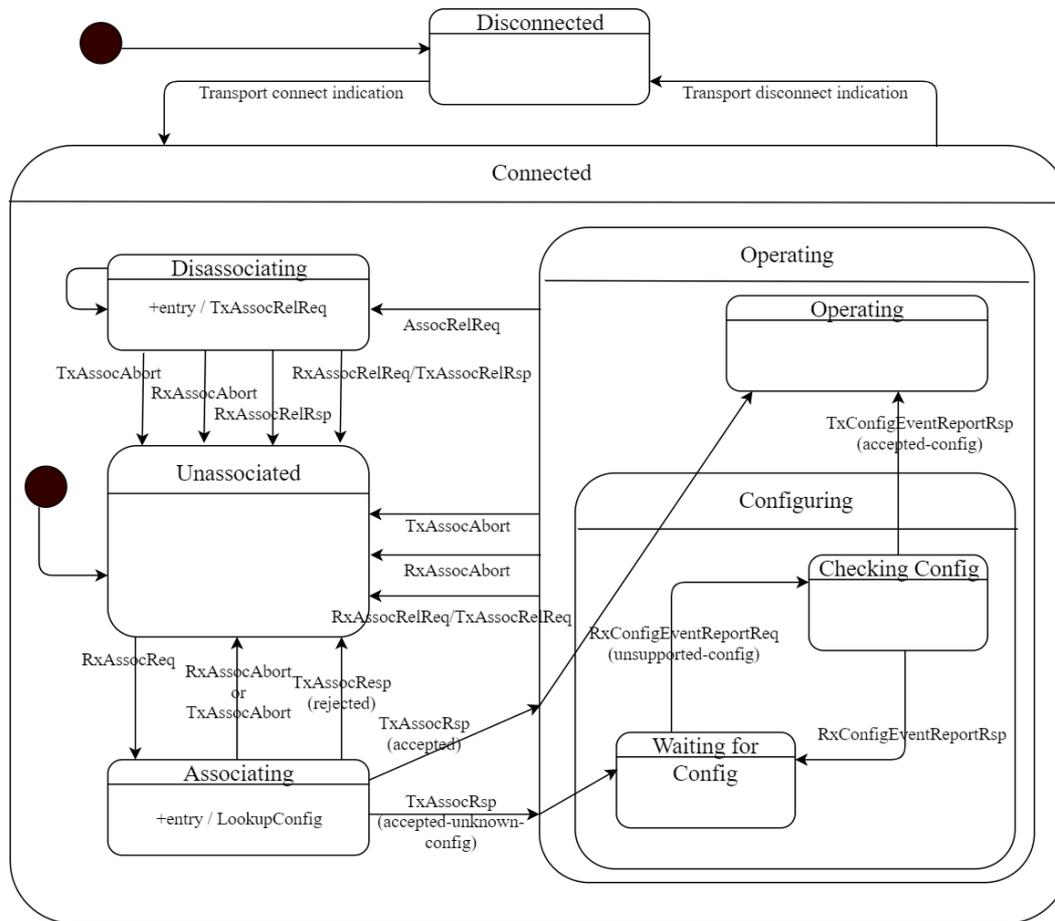


Figura 3.4 – Máquina de Estados de um *Manager* 11073 - imagem adaptada de [48].

Na Figura 3.4 são ilustrados os seguintes estados:

Disconnected (Figura 3.5)

Quando se liga um *Agent* pela primeira vez, este está no estado *Disconnected*, pelo que não foi estabelecida nenhuma ligação entre o *Agent* e *Manager*. É possível voltar a este estado caso uma ligação estabelecida anteriormente seja interrompida.

Connected (Figura 3.5)

Uma vez estabelecida uma ligação entre um *Agent* e um *Manager*, o *Agent* entra no estado *Connected* e permanece enquanto a ligação estiver ativa.

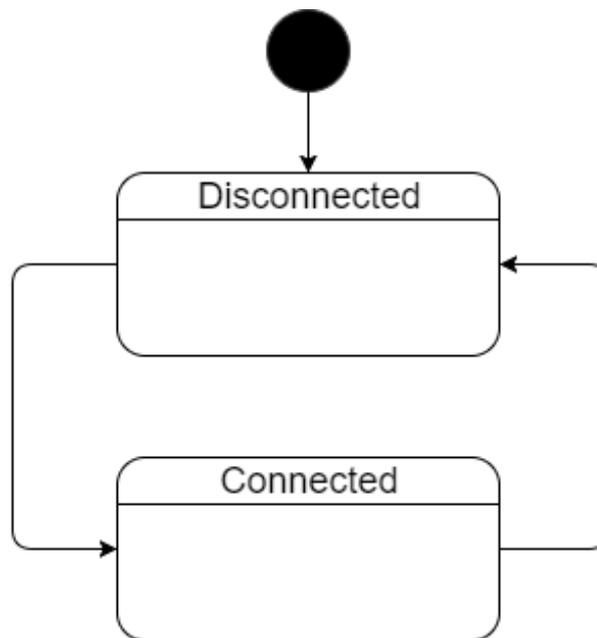


Figura 3.5 – Estado *Connected* e *Disconnected* - imagem adaptada de 3.4.

Unassociated (Figura 3.6)

O *Agent* encontra-se no estado *Unassociated* sempre que não existe uma ligação com o *Manager* ao nível da camada de aplicação. O *Agent* permanece neste estado até iniciar uma associação com o *Manager*. Neste estado pode ocorrer qualquer uma das seguintes situações:

- Uma ligação pode ser estabelecida.
- Um dos dispositivos (*Agent* ou *Manager*) abortou a associação.
- O *Manager* rejeitou o pedido de associação.

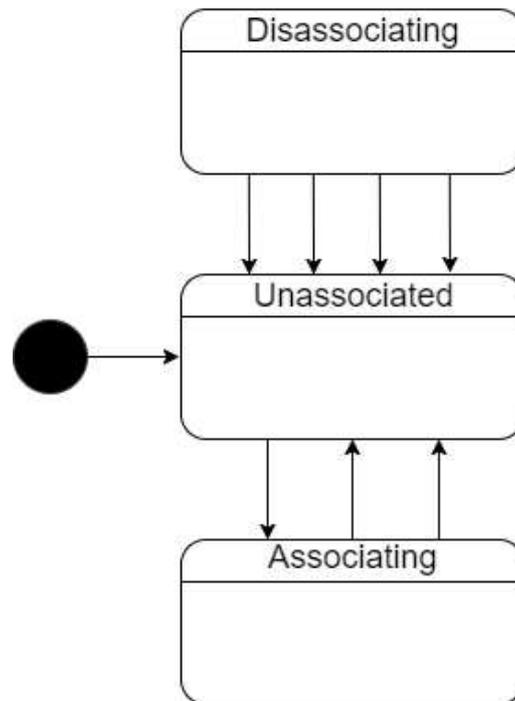


Figura 3.6 – Estado *Unassociated* - imagem adaptada de 3.4.

Associating (Figura 3.7)

Quando o *Agent* faz um pedido de associação, este passa para o estado *Associating* e envia um pedido de associação ao *Manager*.

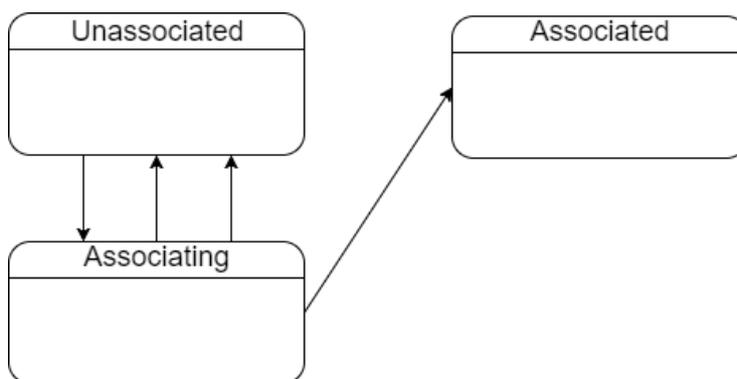


Figura 3.7 – Estado *Associating* - imagem adaptada de 3.4.

Associated (Figura 3.8)

Se o *Manager* e o *Agent* partilharem protocolos e versões de associação comuns, a associação é efetuada e o *Agent* passa para o estado *Associated*. Permanece neste estado até receber um pedido de desconexão ou abortar.

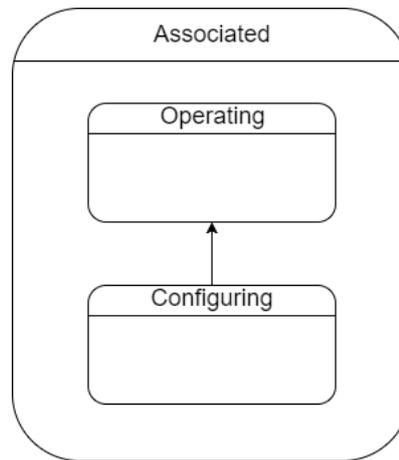


Figura 3.8 – Estado *Associated* - imagem adaptada de 3.4.

Configuring (Figura 3.9)

Quando o *Manager* não reconhece a configuração do *Agent*, este entra no estado *Configuring* e permanece até transmitir a informação da sua configuração ao *Manager* e receber a confirmação do mesmo.

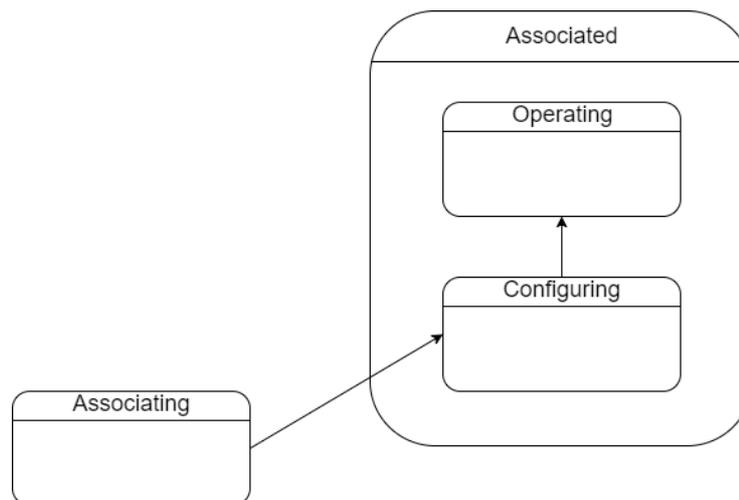


Figura 3.9 – Estado *Configuring* - imagem adaptada de 3.4.

Operating (Figura 3.10)

No estado *Operating* dá-se o início da transferência da informação do *Agent* para o *Manager*. É reconhecida a configuração do *Agent* pelo *Manager*.

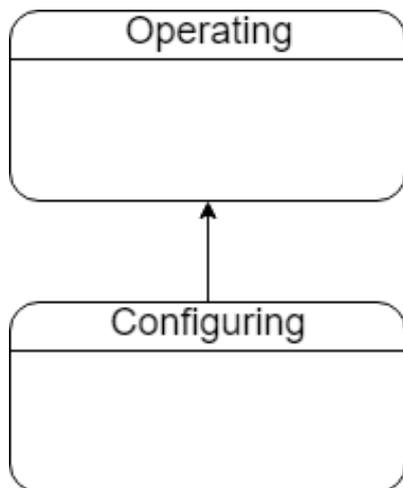


Figura 3.10 – Estado *Operating* - imagem adaptada de 3.4.

Disassociating (Figura 3.11)

Quando o *Agent* encerrar a associação, este entra no estado *Disassociating* e envia um pedido de desassociação ao *Manager*. Se o *Agent* receber a confirmação, este segue para o estado *Unassociated*.

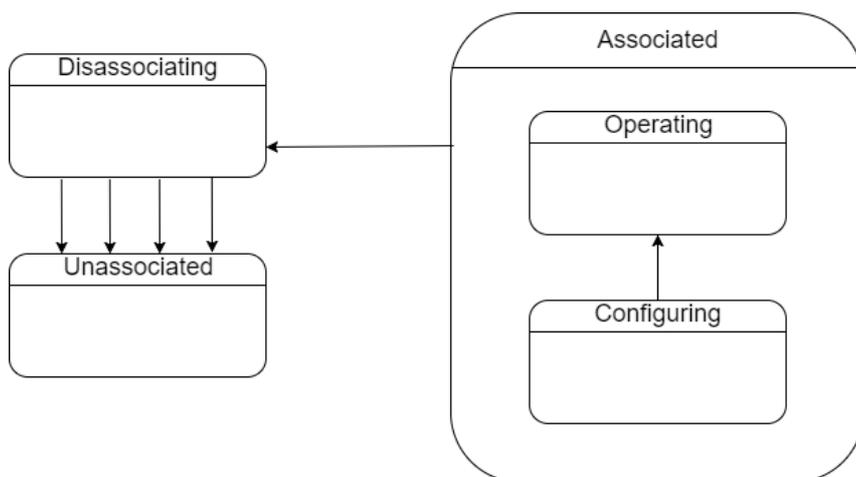


Figura 3.11 – Estado *Disassociating* - imagem adaptada de 3.4.

3.1.6 ASN.1

A *Abstract Syntax Notation 1* (ASN.1) é uma notação que fornece um conjunto de regras formais para descrever a estrutura de dados para a representação, codificação, decodificação e transmissão. Cada objeto e atributo são formalmente definidos usando esta notação. Esta é independente das especificações de codificação do dispositivo em que é utilizada conseguindo assim uma notação que elimina as ambiguidades [49].

A ASN.1 define uma sintaxe para os dados mas não impõe a forma de codificação. Regras para codificações como *XML Encoding Rules* (XER), *Basic Encoding Rules* (BER) ou *Distinguished Encoding Rules* (DER) podem ser utilizadas para codificar dados.

No caso do 11073, os objetos ASN.1 foram convertidos em fluxos de dados binários usando codificações *Medical Device Encoding Rules* (MDER). O MDER é um conjunto otimizado de regras de codificação que suportam um subconjunto de dados primitivos da sintaxe ASN.1. As mensagens médicas são limitadas a estes tipos primitivos que melhor se adequam aos tipos PHD permitindo assim obter uma otimização na codificação [15, 50, 51].

3.2 Bluetooth

Inicialmente pensado pela empresa *Ericson*, em 1994, como forma de eliminar os cabos que ligavam os periféricos dos telemóveis, o Bluetooth é uma tecnologia *standard* sem fios que permite a ligação e troca de dados em curtas distâncias através de ondas rádio de baixa potência de transmissão entre dispositivos próximos uns dos outros, formando assim uma rede pessoal denominada *Personal Area Network* (PAN).

As especificações do Bluetooth foram desenvolvidas e licenciadas pela *Bluetooth Special Interest Group* (BSIG), constituída inicialmente pelas empresas *Nokia*, *Ericson*,

IBM, Toshiba e Intel, facilitando assim o projeto receber mais aceitação pela comunidade tecnológica [52].

3.2.1 Funcionamento Técnico

O Bluetooth utiliza ondas rádio de baixa potência, operando em frequências que vão de 2.4GHz a 2.5GHz, na faixa de frequência conhecida como *Industrial, Scientific and Medical* (ISM). O uso da baixa potência aumenta a economia de energia das baterias e limita-o a um máximo de 100m.

A tecnologia é dividida em quatro classes tendo em conta o alcance das ondas rádio como se verifica na Tabela 3.1:

Tabela 3.1 – Classe dos dispositivos Bluetooth

Classe	Máxima potência permitida		Alcance (m)
	(mW)	(dBm)	
1	100	20	+/- 100 m
2	2.5	4	+/- 10 m
3	1	0	+/- 1 m
4	0.5	-3	+/- 0.5 m

É de notar que os dispositivos possuidores desta tecnologia não necessitam estar na linha de visão um do outro para comunicar. O envio de sinais muito fracos na ordem dos miliWatts é uma das formas de evitar interferências com outros dispositivos.

Durante o processo de comunicação Bluetooth o canal é partilhado com os dispositivos que estão sincronizados, sendo que o dispositivo que fornece os parâmetros de sincronização do relógio é denominado *Master*. Os dispositivos que partilham o relógio do *Master* são designados de *Slaves*. O Bluetooth pode ligar até oito dispositivos simultaneamente, um *Master* e sete *Slaves*, formando uma *piconet*. Cada *piconet* possui apenas um *Master* que funciona como servidor central dos *Slaves* e um *Slave* pode ser *Master* de outra *piconet*. Quando são partilhados elementos de diferentes *piconets* é formada uma *scatternet*. A figura 3.12 ilustra estes conceitos.

É improvável que os dispositivos interfiram entre si uma vez que o Bluetooth usa uma técnica denominada *Spread Spectrum Frequency Hopping* (FHSS) que torna raro um dispositivo transmitir na mesma frequência que outro ao mesmo tempo. Com esta técnica um dispositivo usa aleatoriamente 79 frequências individuais numa determinada faixa de frequência (neste caso ISM) e no caso do Bluetooth as frequências são alteradas até 1600 vezes por segundo, o que faz com que seja bastante improvável um transmissor utilizar o mesmo canal no mesmo tempo que outro transmissor [52–54].

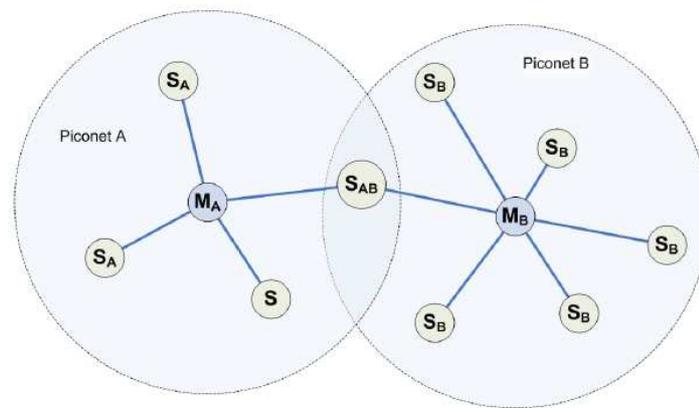


Figura 3.12 – Exemplo de uma *scatternet* [55].

3.2.2 Arquitetura

O Bluetooth possui uma arquitetura protocolar de camadas, composta por [52, 56–60]:

- Protocolos Core - que por sua vez consistem em cinco protocolos, sendo estes:
 - *Bluetooth Radio* - responsável pela modulação/desmodulação dos dados para transmissão e recepção no ar. É basicamente a camada física referindo-se às antenas.
 - *Baseband/Link Controller* - a parte de *Baseband* lida com a sincronização e é responsável pelo formato dos pacotes para transmissão de e para a camada de rádio. A parte do *Link Controller* é responsável por executar

os comandos do *Link Manager* e por estabelecer e manter o *link* estipulado pelo *Link Manager*. Esta camada gere também dois tipos de ligação no Bluetooth, o *Asynchronous Connection Less (ACL)* e o *Synchronous Connection Oriented (SCO)*.

- *Link Manager Protocol (LMP)* - responsável por estabelecer a ligação entre dispositivos Bluetooth e manter a ligação entre eles. Este protocolo também inclui especificações de autenticação e encriptação. A negociação do tamanho dos pacotes entre os dispositivos pode ser tratada por esta camada.
 - *Logical Link Control and Adaptation (L2CAP)* - responsável por adaptar os protocolos da camada superior à camada *BaseBand/Link Controller* e vice versa, fornecendo tanto serviços ACL como serviços SCO. Suporta a multiplexagem de protocolos, uma vez que, protocolos de camadas superiores podem operar por cima do L2CAP (e.g. TCS, SDP, RFCOMM) e também suporta a *Segmentation & Reassembly (SAR)* de forma a que os pacotes de dados que ultrapassem o MTU possam ser segmentados antes de serem transmitidos. Esta funcionalidade e a reversa são executadas pelo L2CAP.
 - *Service Discovery Protocol (SDP)* - manipula informações do dispositivo e dos serviços. As características dos serviços podem ser consultadas para permitir o estabelecimento de uma ligação entre dois ou mais dispositivos. Oferece os meios para descobrir quais os serviços que estão disponíveis pelos outros dispositivos ligados.
- Protocolo de substituição de Cabo - O Bluetooth usa *Radio Frequency Communication (RFCOMM)* como protocolo de substituição de cabo. Este protocolo funciona como uma porta série virtual que permite a transmissão de bits através da emulação de portas na camada L2CAP.
 - Protocolo de Controlo de Telefonia - responsável por definir a sinalização do controlo de chamadas para o estabelecimento de chamadas de voz e dados

entre dispositivos. Este protocolo é apenas utilizado pelo perfil de telefonia sem fios.

- Protocolos Adotados - são protocolos definidos por outras organizações criadoras de *standards* para suporte de interoperabilidade dos produtos *end user*, sendo então incorporados na *stack* protocolar Bluetooth:
 - *Point-to-Point* (PPP) - Na tecnologia Bluetooth este protocolo é projetado para ser executado através do RFCOMM com o objetivo de realizar conexões ponto a ponto.
 - TCP/IP/UDP - protocolos para transmissão de dados, com ou sem fiabilidade, dependendo se for TCP ou UDP.
 - *Object Exchange Protocol* (OBEX) - protocolo para troca de objetos (tramas, dados..) de forma simples e espontânea.

Host Controller Interface (HCI) - O HCI não pertence à arquitetura de camadas. É a interface entre as camadas inferiores e o módulo Bluetooth. Fornece interoperabilidade entre módulos de fabricantes diferentes e contém controlo e registo de eventos.

A Figura 3.13 ilustra os conceitos referidos acima e como se relacionam entre si.

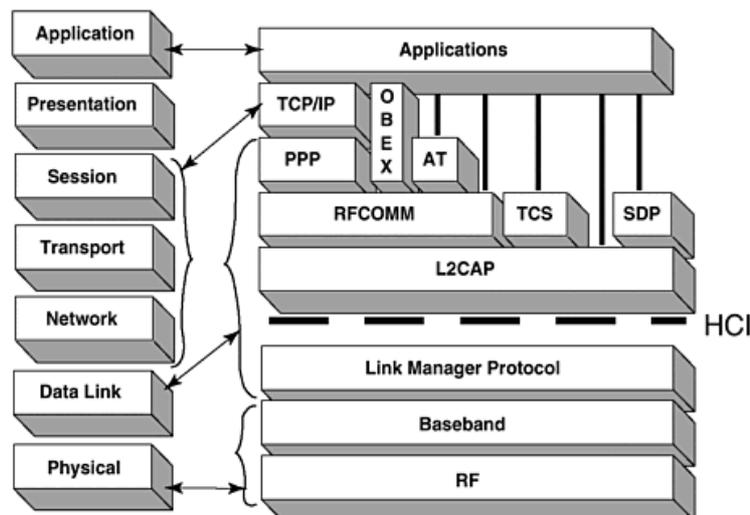


Figura 3.13 – Arquitetura da *Stack* protocolar do Bluetooth [61].

3.2.3 Versões Bluetooth

Desde que o Bluetooth surgiu como tecnologia para transmissão de dados à distância, não pára de expandir as suas características que estão divididas em diversas versões [57, 62, 63]:

- Bluetooth 1.0 e 1.0B - o Bluetooth 1.0 foi publicado em 1998 e foi capaz de transferir os dados numa distância de alguns metros. Foi caracterizado por ter uma fraca interoperabilidade entre produtos de diferentes fabricantes;
- Bluetooth 1.1 e 1.2 - estas atualizações ofereceram suporte para sinais de energia recebidos e aceleraram o processo de deteção e ligação dos dispositivos. O ruído aumentou e a velocidade de transmissão passou a ser cerca de 720 kb/s. Esta versão permite enviar mensagens de voz longas e usar transmissão de áudio e stereo. Possui compatibilidade com a versão anterior.
- Bluetooth 2.0 e 2.1 - lançada em 2001, a versão 2.0 tem como característica a introdução da tecnologia *Enhanced Data Rate* (EDR) que permitiu teoricamente aumentar a velocidade para 3 Mb/s. A versão 2.1 surgiu em 2007 como aperfeiçoamento da anterior, principalmente em termos energéticos, o consumo da energia diminuiu cerca de 10 vezes. Possui compatibilidade com as versões anteriores;
- Bluetooth 3.0 + HS - o Bluetooth 3.0 publicado em 2009 foi pioneiro no uso da transmissão de dados *high speed* (HS). A velocidade aumentou para 24 Mb/s devido à instalação do módulo Bluetooth 2.1 + EDR e do módulo Wi-Fi 802.11. Este último fornece um aumento na velocidade mas não fornece compatibilidade Wi-Fi e tem como desvantagem o aumento do consumo de energia do dispositivo. Possui compatibilidade com as versões anteriores;
- Bluetooth 4.0 e 4.1 - esta versão surgiu em 2010 para corrigir as principais desvantagens da versão anterior - o consumo excessivo de energia. O Bluetooth 4 inclui o módulo Bluetooth tradicional e também o módulo de alta velocidade

com base no protocolo Wi-Fi. Esta versão é usada principalmente em sensores, tais como sensores utilizados na saúde para monitorização dos parâmetros fisiológicos. Em 2013 a versão 4.1 foi introduzida e levou o conceito de *Low Energy* (LE) para outro nível. Permiteu que os dispositivos inteligentes permaneçam conectados durante longos períodos de tempo com gastos mínimos da bateria. Possui compatibilidade com as versões anteriores;

- Bluetooth 5 - lançada oficialmente em junho de 2016, a versão 5 do Bluetooth quadruplica o alcance usando maior potência de transmissão e duplica a velocidade em relação à versão anterior. Apesar do alcance e da velocidade aumentar, o Bluetooth SIG dá a entender que o consumo de energia não será um problema tão grande como em versões anteriores e melhorará novamente os transmissores como foi feito anteriormente. Possui compatibilidade com as versões anteriores.

A velocidade com que o Bluetooth transmite dados é significativamente menor que a do Wi-Fi, em contrapartida, faz menor uso da bateria e permite que os dispositivos se liguem rapidamente uns aos outros sem problemas de ruído/interferência.

3.3 HDP

Os perfis Bluetooth constituem as especificações de interface sem fios para comunicações Bluetooth entre dispositivos. De forma a que os dispositivos possam trabalhar em conjunto é necessário que implementem um perfil comum para esclarecer qual o tipo de dados que o módulo está a transmitir [64, 65].

O *Health Device Profile* (HDP) foi introduzido no Android 4.0 e é um perfil Bluetooth utilizado para criar canais de comunicação entre dispositivos pessoais de saúde e dispositivos de recolha de dados (e.g. *tablet*, *smartphone*, etc.) [15]. Este perfil surgiu para substituir outro perfil com funções semelhantes, o *Serial Port Profile* (SPP), que apenas funciona como substituto do cabo série, não abordando a informação que está a ser transmitida, ao contrário do HDP que fornece uma *framework standard*

para a comunicação da informação médica entre dispositivos.

Este perfil assume vantagens em vários níveis, tais como [6]:

- Médico, cuidados de saúde e *fitness*
O HDP é um perfil especializado, projetado para permitir a interoperabilidade em dispositivos desta área, tornando-o melhor em relação a perfis mais genéricos como o SPP.
- Descoberta de serviços sem fios
O HDP fornece um método de descoberta de serviços sem fios onde está determinado o tipo de dispositivo e as aplicações que ele suporta.
- Comportamento confiável das ligações
O HDP garante um comportamento mais confiável quando o emissor se move além do alcance máximo ou se se desconecta, intencionalmente ou não. Isto permite que o emissor e o recetor reconheçam que a ligação foi interrompida e realizem as ações apropriadas para voltar a estabelecer a ligação. Além disso, os canais de transmissão detetam e reenviam os pacotes corrompidos.
- Canal de controlo robusto
O canal de controlo do HDP requer o Enhanced Retransmission Mode (modo de retransmissão avançado) e o FCS (Frame Check Sequence) para aumentar a confiabilidade do controlo dos canais de comunicação e obter uma operação robusta.
- Suporte para configurações de canais de dados flexíveis
Os canais de dados HDP permitem configurações independentes e flexíveis. Os canais de dados configurados como confiáveis usam Enhanced Retransmission Mode enquanto que os canais de dados configurados como *streaming* usam o modo de *Streaming*. O uso do FCS é opcional.
- Mecanismos eficientes de reconexão
O HDP permite aos dispositivos reter o estado do sistema e eliminar as etapas

de configuração redundantes após uma reconexão. Este procedimento permite que os dispositivos se desconectem quando não há informação nova a receber e que se liguem quando voltam a existir dados para transmitir. Este método reduz o consumo de energia.

O HDP pode ser integrado na pilha Bluetooth tornando mais fácil para os programadores (*developers*) criar aplicações médicas sem fios interoperáveis com outros *softwares* [66].

A Figura 3.14 mostra como se insere o HDP na stack Bluetooth.

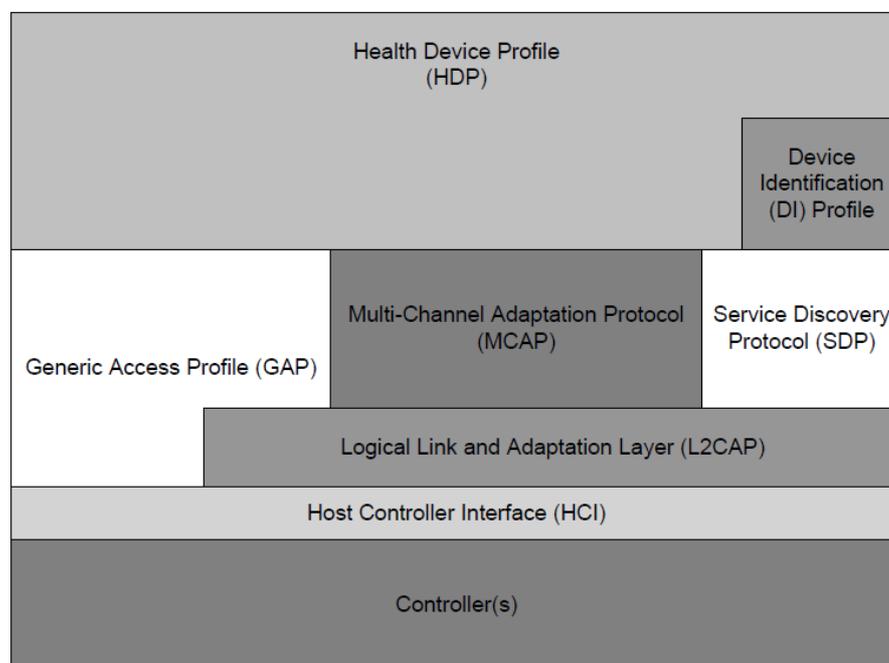


Figura 3.14 – HDP inserido na *Stack* Bluetooth [67].

No contexto do HDP não existem *Agents* e *Managers* mas sim *Sources* e *Sinks* respetivamente, apresentando as mesmas funções que os *Agents* e *Managers* já falados anteriormente. Geralmente, a *Source* é quem inicia a conexão pois é esta que sabe quando tem nova amostras a enviar ao *Sink*.

O HDP tem a capacidade de identificar o tipo de dispositivo usando o *Generic Acces Profile* (GAP) e o SDP. O GAP é responsável por tornar o dispositivo visível

e determinar como dois dispositivos podem (ou não) interagir uns com os outros. O SDP serve para descobrir os dispositivos e os seus serviços disponíveis de modo a saber quais os tipos de dados que suporta [68].

O *Multi-Channel Adaptation Protocol* (MCAP) é o protocolo que define o método a partir do qual é estabelecida a comunicação entre dois dispositivos e é usado pelo HDP para estabelecer e administrar ligações L2CAP. O primeiro canal L2CAP estabelecido entre duas implementações de MCAP é o *Control Channel*, configurado para ser *reliable*, que facilita a criação de *Data Channels*. É obrigatório que exista apenas um *Control Channel*, e no que diz respeito aos *Data Channels* é obrigatório que exista pelo menos um. Os *Data Channels* podem ser configurados de duas formas [69]:

- *Reliable*: canais apropriados para transmitir as medidas feitas pelo dispositivo ou informações de alerta onde é necessário garantir que as medidas/informações cheguem ao destino durante a troca. Qualquer pacote perdido ou corrompido será detetado e retransmitido [67];
- *Streaming*: canais apropriados quando a pontualidade da entrega de cada frame é mais importante do que a entrega segura de cada frame. Os pacotes podem ser descartados se não puderem ser entregues num determinado limite de tempo [67].

4

Interface Android para um Dispositivo Pessoal de Saúde

Neste capítulo apresentam-se os passos efetuados de forma a implementar uma interface Android para o dispositivo pessoal de saúde assim como os resultados obtidos. Por conveniência o dispositivo usado para testes foi um oxímetro certificado pela PCHA (Nonin 9560). Em apêndice à presente dissertação, foram detalhados minuciosamente todos os passos seguidos afim de que este documento possa servir como base de orientação para futuros trabalhos nesta área.

4.1 Materiais e Métodos

Fez-se um levantamento no *google* e na biblioteca de conhecimento *online* (B-on) sobre bibliotecas *open-source* que implementassem a norma ISO/IEEE 11073 usando como palavra chave “library open source 11073”. No seguimento desta pesquisa encontrou-se numa dissertação de mestrado uma biblioteca *open-source* para implementação do *Manager*, designada por *OpenHealth Manager* [15], e em artigos e dissertações diversas encontrou-se a biblioteca Antidote [15, 70–72]. Na data em que se pesquisou, novamente no *google* e no B-on, pelo *OpenHealth Manager*, verificou-se

que toda a informação relativa a este *Manager* já não se encontrava *online*. Em contrapartida, na pesquisa pelo outro *Manager open-source*, o Antidote, encontrou-se alguma informação incluindo um manual técnico que serve de guia para programadores, apesar das últimas atualizações desta biblioteca datarem de 2014.

Uma vez que se verificou, no decorrer deste trabalho, que é possível utilizar a *stack* do Antidote no sistema operativo Android [48], procedeu-se à compilação e instalação desta biblioteca.

Após diversas tentativas de compilação onde se gastou bastante tempo para adaptar a biblioteca aos parâmetros do sistema Android mais recente, conseguiu-se ter sucesso na sua compilação e instalação. Recorde-se que a biblioteca Antidote não sofreu alterações desde 2014, enquanto que neste espaço de tempo houve 16 novas versões do compilador de código nativo associado ao Android *Native Development Kit* (NDK) [73]. O NDK é um conjunto de ferramentas que permitem usar o código C e C++ nas aplicações Android através da criação automática de uma interface em Java, usando o *Java Native Interface* (JNI) [74].

Inicialmente num PC, instalou-se e começou-se a utilizar o *Integrated Development Environment* (IDE) Eclipse, uma vez que as aplicações do Antidote foram desenvolvidas e testadas nesse mesmo ambiente. Mais tarde utilizou-se o Android Studio visto que é o IDE oficial para o desenvolvimento de aplicações Android, tendo que ter sido feita a respetiva migração e adaptação do NDK cuja versão era 14b (março 2017) para a versão 8e (março 2013).

Uma vez instalada a biblioteca foi possível correr pequenas aplicações de teste que comprovaram o correto funcionamento. Na sequência dos testes foi possível desenvolver uma aplicação fazendo uso de um oxímetro que esteve temporariamente disponível para este efeito.

Neste trabalho foram utilizados alguns equipamentos de forma a conseguir fazer a montagem do projeto. O dispositivo de recolha de dados escolhido para se testar as aplicações é um *tablet* Asus MeMo Pad ME302C com Android 4 *Jelly Bean* e Bluetooth 3.0.

4.2 Antidote

A *stack* Antidote é uma implementação do *standard* IEEE 11073-20601, *open-source*, desenvolvido pela Signove e que permite a comunicação com dispositivos pessoais de saúde. Foi desenvolvido com um objetivo muito específico, a portabilidade, de forma a poder ser usado noutras plataformas, apesar de ser testada e utilizada maioritariamente em Linux [48].

O Antidote fornece suporte para transcodificação, isto é, o Antidote interpreta os dados recebidos como sendo dados 11073 mesmo que os dispositivos pessoais de saúde não sigam o *standard* ISO/IEEE 11073-20601. A biblioteca possui poucas dependências externas, é escrita em C e suporta tecnologias de transporte como Bluetooth, USB e TCP/IP [70, 75]. Segundo os autores desta biblioteca, é possível fazer porte do Antidote para o sistema operativo Android.

4.2.1 Componentes do Antidote

Esta secção descreve alguns blocos de componentes (4.1) da *stack* do Antidote.

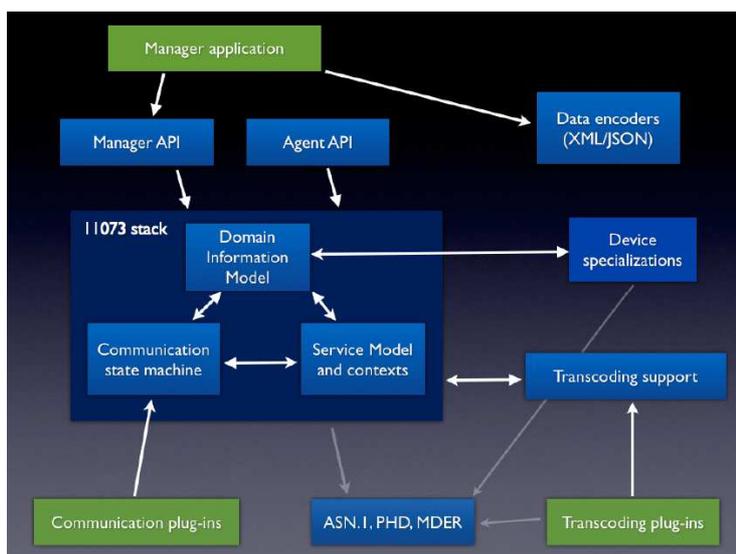


Figura 4.1 – Componentes da *stack* do Antidote.

- *Data Codecs*: A existência de codificadores e decodificadores (*data codecs*) permite que as aplicações não lidem diretamente com a *stack* do IEEE 11073. Estes codificadores e decodificadores são usados pela aplicação e não pela *stack*. O Antidote não obriga à aplicação lidar diretamente com os tipos de MDS como medições e atributos MDS, ao invés, o Antidote encapsula cada dado em estruturas *DataList*. Cada *DataList* é enviada via *Inter Process Communication* (IPC) e necessita de ser codificada. Antecipando esta necessidade, o Antidote dispõe de codificadores para XML e JSON, formatos usados na maioria das linguagens de programação. Por outro lado, o Antidote não impõe barreiras, pelo que é possível aceder aos atributos MDS pela aplicação se tal for preciso [48].

- *Transport plug-ins*: O Antidote é independente do protocolo de transporte mas, num determinado momento, é necessário que a aplicação faça interface com a camada de transporte. Nesta etapa é chamado o *plug-in* correspondente ao protocolo escolhido (USB ou Bluetooth) para fazer interface. Um *plug-in* de transporte deve fornecer os serviços seguintes [48]:
 - notificar quando uma conexão é aberta ou fechada;
 - criar e manter um identificador único para cada dispositivo ligado;
 - terminar a ligação quando é pedido;
 - notificar a chegada de um novo APDU;
 - aceitar um APDU vindo da *stack* e enviá-lo;
 - iniciar e terminar temporizadores assíncronos;
 - proteger uma ligação no caso de usar múltiplos *threads*.

Os restantes componentes foram abordados no Capítulo 3.

4.3 Implementação usando o Antidote

Para se fazer uso da biblioteca Antidote teve de se instalar previamente o Android Studio com a versão 8e do compilador de código nativo, o NDK, e proceder à migração, parcialmente manual, do projeto originalmente formatado para o IDE Eclipse. Este processo de migração inclui inserir os ficheiros pedidos nas mensagens do sumário do *import*, nas diretorias correspondentes e criar uma pasta com o nome “antidote” composta de todos os ficheiros do “antidote-master” descarregado, exceptuando a pasta “Android Samples”. De seguida, teve de se criar um ficheiro com o nome “Application.mk” com o conteúdo `APP_PLATFORM:=android-24`. Depois de se configurar o Android Studio e seleccionar os caminhos corretos das diretorias, fez-se a montagem do projeto com o comando `ndk-build NDK_DEBUG=1`. No apêndice está descrita a implementação minuciosa dos passos seguidos.

Até esta fase configurou-se a parte referente ao *Manager*, sendo necessário para efetuar testes recorrer a um *Agent* capaz de enviar dados e poderem ser lidos no *tablet*. O manual técnico do Antidote [48] explicita que existem duas formas de testar as aplicações, uma usando uma *applet* para fazer testes 11073 que simula o envio de dados e outra é usando um dispositivo pessoal de saúde com HDP. Inicialmente optou-se por utilizar a *applet* por não se ter acesso a nenhum dispositivo pessoal de saúde com Bluetooth e perfil HDP. Uma vez que esta *applet* apenas funciona em ambientes Linux, instalou-se o Antidote no sistema operativo Ubuntu. Feito isto, decidiu-se testar a *applet* com as duas aplicações Android e apesar de executar os passos indicados no manual foi impossível visualizar qualquer resultado.

Posteriormente conseguiu-se ter acesso a um oxímetro Nonin 9560, um dispositivo pessoal de saúde com tecnologia Bluetooth e suporte para HDP, certificado pela Continua Alliance. Os dados deste dispositivo podem ser enviados por Bluetooth para dispositivos móveis como *smartphone* e *tablet* que usem uma aplicação que implemente a norma ISO/IEEE 11073. A tabela 4.1 descreve as características técnicas mais relevantes.

Tabela 4.1 – Resumo das características técnicas do oxímetro Nonin 9560.

Saturação do oxigénio	0% a 100%
Batimentos cardíacos por minuto	18 a 321
Alimentação	Duas pilhas 1.5V tamanho AAA
Sessões de operação	até 600
Bluetooth	Versão 2.0, classe 1
Perfis Bluetooth	SPP e HDP

Nesta situação o oxímetro será o *Agent* e o *tablet* será o *Manager*. Depois do oxímetro estar emparelhado com o *tablet*, isto é, inserindo um *pin* fixo que o oxímetro possui escrito na sua lateral, de modo a que se estabeleça uma ligação segura em que ambos os dispositivos se “conhecem”, é necessário que se abram as aplicações Android na seguinte ordem: em primeiro a aplicação HealthService (abre um ecrã) e de seguida abrir a aplicação que irá funcionar como *user interface* (UI) baseada na HealthServiceTest fornecida na biblioteca Antidote. Feito isto, basta inserir o dedo no oxímetro e esperar que a ligação seja estabelecida. A Figura 4.2 demonstra o aspeto físico do sistema montado.



Figura 4.2 – Esquema de montagem do sistema proposto.

4.3.1 Implementação de uma Interface do Utilizador

Uma vez testada a funcionalidade, desenvolveu-se a interface do utilizador adequada a um utilizador final. O Android Studio possui um editor de *layout* em XML, sendo este exibido sempre que se abre um ficheiro de *layout* XML e é composto por diversas ferramentas como a *Palette*, *Component Tree* e *Toolbar*. O XML, um subconjunto da *Standard Generalized Markup Language* (SGML), é uma linguagem de marcação para documentos contendo informações estruturadas através de *tags*. A Figura 4.3 mostra um excerto utilizado na UI deste estudo e a Figura 4.4 e a Figura 4.5 ilustram a UI na sua fase final.

```
89
90 <LinearLayout
91     android:layout_width="match_parent"
92     android:layout_height="match_parent"
93     android:layout_weight="1.5"
94     android:orientation="vertical"
95     android:layout_marginTop="90dp">
96
97     <LinearLayout
98         android:layout_width="match_parent"
99         android:layout_height="wrap_content"
100         android:orientation="horizontal">
101
102         <TextView
103             android:layout_width="match_parent"
104             android:layout_height="match_parent"
105             android:text="%SpO2:"
106             android:gravity="right"
107             android:paddingRight="15dp"
108             android:textColor="@android:color/black"
109             android:textSize="40dp"
110             android:textStyle="bold"
111             android:layout_weight="1" />
112
113         <TextView
114             android:layout_width="match_parent"
115             android:layout_height="match_parent"
116             android:paddingLeft="15dp"
117             android:textColor="@android:color/black"
118             android:textSize="40dp"
119             android:layout_weight="1"/>
120     </LinearLayout>
121 </LinearLayout>
```

Figura 4.3 – Excerto de código do XML.



Figura 4.4 – Menu inicial da aplicação elaborada.

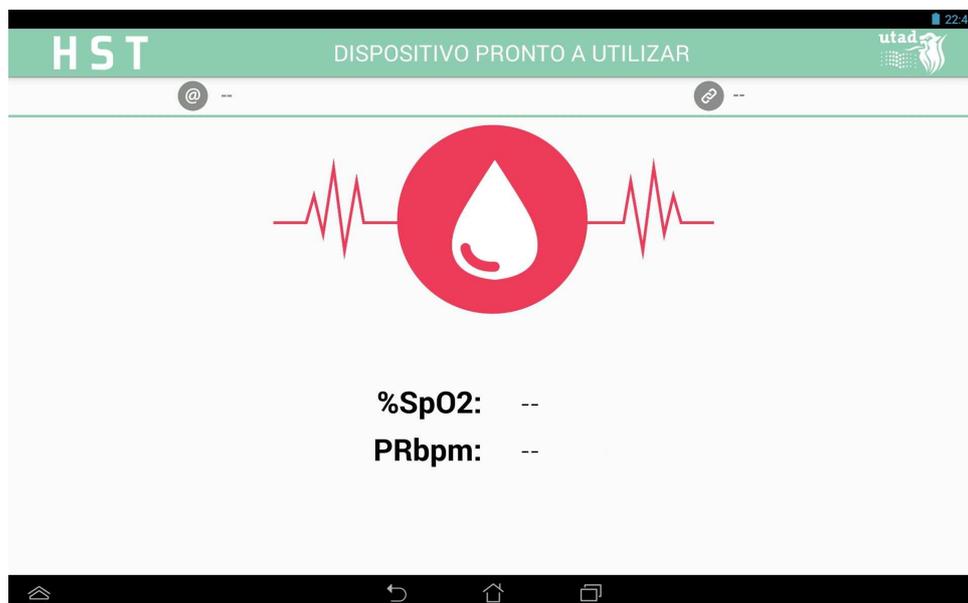


Figura 4.5 – Menu do oxímetro da aplicação elaborada.

4.4 Resultados Obtidos

O manual técnico do Antidote indicava que as duas aplicações se encontravam no *gitorious.org*, o que não se verificou, pois ao aceder a essa página era afixada a mensagem que não existiam projetos. Após alguma pesquisa depressa se encontraram no *github.com* e no *oss.signove.com* com o nome “antidote-master”.

Depois das aplicações serem abertas e corrigidas adequadamente no IDE Eclipse, começou-se a ter um erro ao compilar a aplicação *HealthServiceTest*. Este erro era devido à falta de uma invocação aos métodos abstratos *public void ReleaseAssociation(String dev)* e *public void AbortAssociation(String dev)* na classe *HealthServiceTestActivity.java*. Depois de ultrapassar este problema, esta não abria no *tablet*. Concluiu-se que tal se devia ao facto da compilação no IDE Eclipse ser realizada através do ADT Plugin (Android Developer Tools), e atualmente este já não é suportada pelos dispositivos Android [76], optando-se então por usar o IDE oficial de Android.

No IDE Android Studio foram seguidos os passos referenciados em apêndice (Apêndice A), tendo sido obtido sucesso na compilação e instalação após diversas tentativas como se descreve de seguida:

- utilizar a versão 8e do compilador de código nativo NDK, visto que a versão atual durante o projeto (14b) apresentava sempre a mensagem de erro “*linker commmand failed with exit code 1*”, que apesar de bastante pesquisa não foi possível resolvê-lo, sendo necessário recorrer à versão utilizada quando o Antidote surgiu ao público;
- adicionar em `AndroidStudioProjects\HealthService1\HealthService\src\main` as pastas e ficheiros `aidl\com\signove\health\service\HealthAgentAPI.aidl` e `aidl\com\signove\health\service\HealthServiceAPI.aidl`;
- acrescentar em `AndroidStudioProjects\HealthService1\HealthService\src\main\jni\Android.mk` a linha: `APP_ALLOW_MISSING_DEPS=true`;

- em `AndroidStudioProjects\HealthService1\HealthService\src\main\jni` criar um ficheiro com o nome “Application.mk” com o conteúdo:
`APP_PLATFORM := android-24.`

O uso da *applet* de teste incluída na biblioteca Antidote não gerou os resultados esperados, anunciados no manual. Um dos motivos que poderá justificar isto é o facto de o ambiente Linux onde se instalou esta aplicação estar a correr numa máquina virtual sobre o Windows 10.

Depois de se ter acesso a um oxímetro Nonin, certificado pela PCHA, já foi possível testar as aplicações com este dispositivo pessoal de saúde e confirmar o correto funcionamento das mesmas. O resultado obtido da uma medição na fase de testes pode ser observado na Figura 4.6.

Em determinadas alturas verificou-se que a ligação entre o oxímetro e as aplicações *HealthService* e *HealthServiceTest* não era efetuada. Após pesquisa concluiu-se que o oxímetro Nonin 9560 muitas vezes procura por uma ligação série. A solução encontrada para resolver este caso é refazer o emparelhamento entre o oxímetro e o *tablet*, de forma a que o oxímetro consiga ver um serviço HDP e repetir o mesmo processo caso a ligação volte a não acontecer. Esta solução, embora primitiva, foi apontada numa dissertação de mestrado [72], onde é referido que foi conseguida em conjunto com um dos autores da biblioteca Antidote e que neste trabalho se comprovou.



Figura 4.6 – Medição da percentagem de oxigénio presente no sangue e dos batimentos cardíacos por minuto.

5

Conclusões e trabalho futuro

No âmbito do presente trabalho, houve necessidade de efetuar pesquisa bibliográfica com alguma profundidade em várias áreas técnicas. Começou-se pelo estudo da norma ISO/IEEE 11073 PHD e como esta se implementaria num sistema Android, seguidamente pesquisou-se sobre a temática das comunicações sem fios, mais especificamente o Bluetooth e o seu perfil próprio para dispositivos pessoais de saúde.

O trabalho desenvolvido consistiu na implementação de uma aplicação Android apta para comunicar com dispositivos pessoais de saúde, normalizando os dados recebidos para que futuramente possam ser disponibilizados de forma normalizada a outras aplicações de saúde ou até a bases de dados como os registos eletrónicos de saúde. A implementação da biblioteca *open-source* Antidote assim como as duas aplicações de teste fornecidas pela Signove foi importante para garantir os fundamentos para a construção de aplicações Android baseadas na norma ISO/IEEE 11073. Contudo foi necessário fazer várias adaptações quer a nível da organização do projeto, quer ao nível do contexto Android de forma a conseguir a correta compilação e funcionalidade com o oxímetro certificado pela Continua Alliance usado nos testes.

A utilização deste sistema em ambientes hospitalares seria de grande importância visto que se evitariam erros humanos na inserção das medições na base de dados e

far-se-ia uma melhor gestão do tempo, permitindo uma maior eficiência de trabalho. Os objetivos para este trabalho foram cumpridos. A biblioteca *open-source* Antidote, sendo a única encontrada funcional, foi uma mais valia para se comprovar a comunicação da informação segundo a norma ISO/IEEE 11073 e assim foi possível criar uma UI adequada aos dispositivos pessoais de saúde.

Para trabalho futuro poder-se-ia unir as duas aplicações numa só obtendo assim uma solução mais elegante. Atualmente o Antidote está limitado a quatro especializações e para abranger mais especializações seria necessário uma atualização da biblioteca. Seria de valor salvaguardar as medições efetuadas no próprio dispositivo de recolha de dados, se tal fosse desejado, e seguidamente enviar aos serviços de saúde. Isto exigirá a implementação das restantes interfaces da PCHA e assim encaminhar a informação até aos registos eletrónicos de saúde.

Também poder-se-ia fazer um teste de usabilidade da interface do utilizador por forma a perceber se esta é suficientemente intuitiva para que o utilizador final a use sem esforço e perceba o funcionamento da interface, assim como interprete corretamente os resultados obtidos.

Referências bibliográficas

- [1] Saude Bem Estar. Qualidade de vida. <https://www.saudebemestar.pt/pt/blog-saude/qualidade-de-vida/>. Acedido a 16-Junho-2017. 1
- [2] World Health Organization. What is the who definition of health. <http://www.who.int/suggestions/faq/en/>. Acedido a 10-Junho-2017. 1
- [3] Maria Isabel Fernandes Praça. Qualidade de vida relacionada com a saúde: a perspectiva dos utentes que frequentam os centros de saúde o aces trás-os-montes — nordeste. Master's thesis, Instituto Politécnico de Bragança, 2012. 1
- [4] Rui Nunes and Guilhermina Rego. *Prioridades na Saúde*. Mc Graw-Hill, 2002. 1
- [5] Significados. Significado de qualidade de vida. <https://www.significados.com.br/qualidade-de-vida/>. Acedido a 16-Junho-2017. 1
- [6] Bluetooth SIG. *HDP Implementation Guidance Whitepaper*, 2009. 2, 38
- [7] Signove. Ieee 11073 stack. <https://github.com/signove/antidote>, . Acedido a 15-Junho-2017. 3

- [8] Department of Health. What is a medical device. <https://www.tga.gov.au/what-medical-device>. Acedido a 19-Junho-2017. 8
- [9] Infarmed. Dispositivos médicos. <http://www.infarmed.pt/web/infarmed/entidades/dispositivos-medicos>. Acedido a 19-Junho-2017. 8
- [10] Medical Expo. <http://www.medicalexpo.com/pt/product-d-company-limited/product-67468-622809.html>. 8
- [11] Turner Medical. https://www.turnermedical.com/NONIN_9560_ONYX_2_BLUETOOTH_WIRELESS_FINGER_PULSE_p/nonin_9560_onyx_2.htm.
- [12] Brian Dolan. <http://www.mobihealthnews.com/3676/continua-certifies-bluetooth-enabled-bp-monitor-weight-scale>.
- [13] Philips. https://www.usa.philips.com/c-p/DL8740_37/ear-thermometer. 8
- [14] U.S. Government. Health information technologies: How innovation benefits patients. <https://www.gpo.gov/fdsys/pkg/CHRG-113hhr80806/html/CHRG-113hhr80806.htm>. Acedido a 23-Junho-2017. 9
- [15] Ricardo Miguel Carneiro Faria. Middleware para comunicação com dispositivos médicos. Master's thesis, Faculdade de Ciências da Universidade do Porto, 2015. 9, 19, 22, 31, 37, 41
- [16] apdsi. Interoperabilidade. <http://www.apdsi.pt/glossary/359/121>. Acedido a 23-Junho-2017. 10
- [17] GDT Interop. Definition of interoperability. <http://interoperability-definition.info/en/>. Acedido a 23-Junho-2017. 10
- [18] Associação Portuguesa de Segurança. A normalização em portugal. <https://www.apsei.org.pt/normalizacao/a-normalizacao-em-portugal/>. Acedido a 24-Junho-2017. 10

- [19] IPQ. *Formação sobre Normalização para Comissões Técnicas*. 10
- [20] Electronic Design. What is the difference between de jure and de facto standards? <http://www.electronicdesign.com/embedded/what-s-difference-between-de-jure-and-de-facto-standards>. Acedido a 24-Junho-2017. 11
- [21] ISO. International organization for standardization - about us. <https://www.iso.org/about-us.html>, . Acedido a 24-Junho-2017. 11
- [22] CEN. European committee for standardization - who we are. <https://www.cen.eu/about/Pages/default.aspx>, . Acedido a 24-Junho-2017. 11
- [23] CEN. European committee for standardization - our role in europe. <https://www.cen.eu/about/RoleEurope/Pages/default.aspx>, . Acedido a 24-Junho-2017. 11
- [24] ANSI. Introduction to ansi. https://www.ansi.org/about_ansi/introduction/introduction?menuid=1. Acedido a 24-Junho-2017. 11
- [25] infopedia. Ansi. [https://www.infopedia.pt/\\$ansi](https://www.infopedia.pt/$ansi). Acedido a 24-Junho-2017. 11
- [26] cmtecnologia. Interoperabilidade na saúde. <https://www.cmtecnologia.com.br/tecnologia/interoperabilidade-em-saude/>. Acedido a 23-Junho-2017. 12
- [27] IHE Portugal. Ihe portugal. <http://www.ihe.pt/index.php>. Acedido a 25-Junho-2017. 12
- [28] AprendIS. Ihe. <http://aprendis.gim.med.up.pt/index.php/IHE>, . Acedido a 26-Junho-2017.
- [29] IHE. Integrating the healthcare enterprise (ihe). <https://www.ihe.net/>. Acedido a 26-Junho-2017. 12

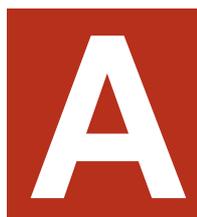
- [30] AprendIS. Digital imaging and communications in medicine. <http://aprendis.gim.med.up.pt/index.php/DICOM>, . Acedido a 20-Junho-2017. 12
- [31] AprendIS. <http://aprendis.gim.med.up.pt/index.php/HL7>, . Acedido a 20-Junho-2017. 13
- [32] HL7. Health level seven international. <http://www.hl7.org/about/index.cfm?ref=common>. Acedido a 20-Junho-2017. 13
- [33] ClinicalTrialsArena. Embracing personal connected health as a novel way to managing patient generated data. <http://www.clinicaltrialsarena.com/news/>. Acedido a 13-Abril-2017. 13
- [34] Chuck Parker. Nfc in personal connected health. <https://www.slideshare.net/NFC-Forum/continua-nfc-forum-developer-spotlight-2014>. Acedido a 3-Setembro-2017. 13
- [35] ITU-T. *HSTP-H810-XCHF Fundamentals of data exchange within ITU-T H.810 Continua Design Guideline architecture*. 14, 15
- [36] Frank Wartena, Johan Muskens, Lars Schmitt, and Milan Petkovic. Continua: The reference architecture of a personal telehealth ecosystem. In *e-Health Networking Applications and Services (Healthcom)*, 2010. 15
- [37] Continua. *Continua H.810 Interoperability design guidelines for personal health systems*. 15
- [38] Personal Connected Health Alliance. Continua design guidelines. <http://www.pchalliance.org/continua-design-guidelines>. Acedido a 13-Abril-2017. 15
- [39] HIMSS. Personal connected health alliance (pcha) celebrates 10 years of continua design guidelines, the only open framework for interoperability of personal connected health devices and solutions. <http://www.himss.org/news/personal-connected-health-alliance-pcha-celebrates-10-years-continua-design-guidelines>. Acedido a 13-Abril-2017. 15

- [40] ISO. Iso/ieec 11073. <http://www.iso.org/iso/search.htm?qt=11073&searchSubmit=Search&sort=rel&type=simple&published=true>, . Acedido a 30-Junho-2017. 17
- [41] Edward H. Shortliffe and James J. Cimino. *Biomedical Informatics: Computer Applications in Health Care and Biomedicine*. Springer, 2014. 18
- [42] IEEE. *Health Informatics - Personal health device communication - Part 00103: Overview*, 2012. 18, 19, 23, 25
- [43] IEEE 11073. Iso/ieec 11073 personal health data (phd) standards. <http://11073.org/>, 2017. Acedido a 30-Junho-2017. 18, 21, 22, 23
- [44] Santiago Led, Leyre Azpilicueta, Miguel Martinez-Espronceda, Luis Serrano, and Francisco Falcone. Operation, analysis and optimization of wireless sensor devices in health oriented monitoring systems, 01 2015. 20, 21
- [45] Mário Jorge Marques da Silva. Dispositivos móveis na recolha eletrónica de dados em estudos clínicos. Master's thesis, Universidade de Aveiro, 2014. 21, 22
- [46] Konstantina S. Nikita. *Handbook of Biomedical Telemetry*. Wiley-IEEE Press, 2014. 23
- [47] openi. Development of cell phone application for blood glucose self-monitoring based on iso/ieec 11073 and hl7 ccd. https://openi.nlm.nih.gov/detailedresult.php?img=PMC4434067_hir-21-83-g002&req=4. Acedido a 2-Julho-2017. 25
- [48] Signove. *Antidote: Program Guide*, . 25, 26, 42, 43, 44, 45
- [49] Liane Margarida Rockenbach Tarouco. A notação usada em asn.1. <http://penta.ufrgs.br/rc952/asn1/asn1b1.html>. Acedido a 12-Julho-2017. 31
- [50] IBM. Primitive asn.1 data types. https://www.ibm.com/support/knowledgecenter/en/SSLTBW_2.1.0/com.ibm.zos.v2r1.istagt0/prmtyp.htm. Acedido a 13-Julho-2017. 31

- [51] Kuinam J.Kim and Kyung-Yong Chung. *IT Convergence and Security 2012*. Springer, 2012. 31
- [52] DevMedia. Introdução à comunicação bluetooth no android. <http://www.devmedia.com.br/introducao-a-comunicacao-bluetooth-no-android/27636#>. Acedido a 3-Julho-2017. 32, 33
- [53] Curt Franklin and Julia Layton. How bluetooth works. <http://electronics.howstuffworks.com/bluetooth2.htm>. Acedido a 5-Julho-2017.
- [54] TechTarget. frequency-hopping spread spectrum. <http://searchnetworking.techtarget.com/definition/frequency-hopping-spread-spectrum>. Acedido a 6-Julho-2017. 33
- [55] Prof Lothar Thiele, Matthias Dyer, and Jan Beutel. Role and link-state selection for bluetooth scatternets. oct 2017. 33
- [56] RF Wireless World. Bluetooth protocol stack/layers. <http://www.rfwireless-world.com/Tutorials/Bluetooth-protocol-stack.html>, . Acedido a 6-Julho-2017. 33
- [57] Bluetooth SIG. Bluetooth. <https://www.bluetooth.com/marketing-branding/brand-enforcement-program>, . Acedido a 6-Julho-2017. 36
- [58] Tutorial-Reports. Bluetooth tutorial. <http://www.tutorial-reports.com/wireless/bluetooth/adoptedprotocols.php>. Acedido a 6-Julho-2017.
- [59] Sonali Parab. Protocols in bluetooth. <https://pt.slideshare.net/sonalizoya/protocols-in-bluetooth-32174066>. Acedido a 6-Julho-2017.
- [60] Rajat Malhotra. Bluetooth technology. <https://www.slideshare.net/rajatmal4/bluetooth>. Acedido a 6-Julho-2017. 33
- [61] Basavaraj Patil, Yousuf Saifullah, and Stefano Faccin. Bluetooth protocol. <http://flylib.com/books/en/4.215.1.116/1/>. 35

- [62] RF Wireless World. Difference between bluetooth 5 and bluetooth 4.2 — bluetooth 5 vs bluetooth 4.2 feature comparison. <http://www.rfwireless-world.com/Terminology/Bluetooth-5-vs-bluetooth-4-2.html>, . Acedido a 7-Julho-2017. 36
- [63] Bluetooth SIG. Bluetooth 5 quadruples range, doubles speed, increases data broadcasting capacity by 800%. <https://www.bluetooth.com/news/pressreleases/2016/06/16/-bluetooth5-quadruples-rangedoubles-speedincreases-data-broadcasting-capacity>, . Acedido a 7-Julho-2017. 36
- [64] Developers. Bluetooth. <https://developer.android.com/guide/topics/connectivity/bluetooth.html?hl=pt-br>, . Acedido a 22-Julho-2017. 37
- [65] Bluetooth SIG. Traditional profile specifications. <https://www.bluetooth.com/specifications/profiles-overview>, . Acedido a 22-Julho-2017. 37
- [66] Len Ott. *The Evolution of Bluetooth in Wireless Medical Devices*. 39
- [67] Bluetooth SIG. *Health Device Profile*, . 39, 40
- [68] adafruit. Introduction to bluetooth low energy. <https://learn.adafruit.com/introduction-to-bluetooth-low-energy/gap>. Acedido a 23-Julho-2017. 40
- [69] Bluetooth SIG. *Multi-Channel Adaptation Protocol*, . 40
- [70] Aravind S. Nair and V. Natarajan. Developing software for ieee 11073 device specializations using antidote library. *International Journal of Engineering and Innovative Technology (IJEIT)*, 5, apr 2016. 41, 43
- [71] Diogo Miguel das Neves Francisco. Tecnologias bluetooth para ambient assisted living (aal). Master’s thesis, Universidade do Minho, 2014.
- [72] Sílvio Manuel Bastos Dias. Aplicações móveis na monitorização de dados biométricos na dpoc. Master’s thesis, Universidade de Aveiro, 2014. 41, 50

- [73] Developers. Ndk revision history. https://developer.android.com/ndk/downloads/revision_history.html, . Acedido a 5-Maio-2017. 42
- [74] Developers. Primeiros passos com o ndk. <https://developer.android.com/ndk/guides/index.html>, . Acedido a 4-Maio-2017. 42
- [75] Elvis Pfutzenteuter. Antidote, a biblioteca ieee 11073 de código aberto. https://epxx.co/logbook/entries/antidote_pt.html. Acedido a 29-Julho-2017. 43
- [76] Developers. Adt plugin (unsupported). <https://developer.android.com/studio/tools/sdk/eclipse-adt.html>, . Acedido a 15-Junho-2017. 49



Implementação detalhada

Implementação detalhada

Após a instalação do Android Studio estar concluída, seguiram-se as seguintes etapas:

1. Download do *package* *Android Native Development Kit* (NDK) versão 8e.
2. Teve de se definir o caminho para a localização do pacote NDK de modo a que futuramente seja possível a sua compilação.
3. Fez-se o *import* das duas aplicações feitas em Eclipse para o Android Studio;
4. Após fazer o *import* surgiu-nos um sumário sobre o mesmo a dizer que nem todos os ficheiros foram “importados”. Teve de se mover manualmente para o novo projeto as pastas e ficheiro `aidl\com\signove\health\service\HealthAgentAPI.aidl` e `aidl\com\signove\health\service\HealthServiceAPI.aidl`, para a diretoria `AndroidStudioProjects\HealthService1\HealthService\src\main`;

5. Também se teve de adicionar manualmente ficheiros `HealthAgentAPI.aidl` e `HealthServiceAPI.aidl` para a diretoria `AndroidStudioProjects\HealthService1\HealthService\src\main\java\com\signove\health\service` uma vez que era apresentado um erro que referia a falta destas duas APIs neste caminho;
6. Seguindo o manual guia, teve de se ir à diretoria `AndroidStudioProjects\HealthService1\HealthService\src\main\jni` criar uma pasta “antidote” e copiar para essa mesma tudo o que está na pasta “antidote-master” excetuando a “Android-Samples”;
7. Dentro de `jni\antidote\src` teve de se acrescentar o ficheiro "`healthd_android.c`" copiado de `antidote-master\apps`;
8. Abrindo o ficheiro “`Android.mk`” que se encontra em `AndroidStudioProjects\HealthService1\HealthService\src\main\jni`, teve-se de acrescentar a linha: `APP_ALLOW_MISSING_DEPS=true`. Esta linha corrigiu um problema de dependências que fazia com que a compilação parasse;
9. Na diretoria `AndroidStudioProjects\HealthService1\HealthService\src\main\jni` teve de se criar um ficheiro “`Application.mk`” com o conteúdo:
`APP_PLATFORM := android-24;`

Feito isto, configuraram-se algumas definições do Android Studio para a compilação dos exemplos:

1. Teve de se ir a Gradle Scripts - `gradle-wrapper.properties` e adicionar a linha de código seguinte: `android.useDeprecatedNdk=true`. Isto porque dava um erro indicando que o projeto migrado possui um script de compilação que usa o antigo suporte NDK.
2. No menu da esquerda, teve de se clicar em cima de “`cpp-antidote`” e fazer “`Link to C++ Project with Gradle`” e escolheu-se “`ndk-build`”. De seguida selecionou-se o caminho do “`Android.mk`” que está em `AndroidStudioProjects\HealthService1\HealthService\src\main\jni`.

3. Compilar pelo terminal a aplicação HealthService usando o comando `D:\Sdk\ndk-bundle\ndk-buildNDK_DEBUG=1`.
4. Após a parte de código nativo estar pronta, teve de se instalar as aplicações no *tablet*.

