



WISE-MANager: Criação e Gestão de Sessões Colaborativas em Redes de Sensores sem Fio

João Filipe Freitas Santos

(Licenciado)

*Tese Submetida à Universidade da Madeira para a
Obtenção do Grau de Mestre em Engenharia Informática*

Funchal – Portugal

Novembro 2010

Orientador:

Professora Doutora Laura Margarita Rodríguez Peralta

*Professor Auxiliar do Centro de Ciências Exactas e da Engenharia da Universidade da
Madeira*

AGRADECIMENTOS

No decorrer desta dissertação de mestrado recebi diversos incentivos de forma a concluir esta tese. Gostaria de agradecer à minha orientadora, Prof. Dr^a Laura Rodriguez pela paciência e disponibilidade que sempre demonstrou ao longo deste processo.

Gostaria de agradecer também à minha família e à minha namorada Fátima pelo apoio e motivação dados ao longo da realização deste projecto de mestrado.

Gostaria também de agradecer aos restantes elementos do projecto *WISE-MUSE*, o Bruno Gouveia, Dário Fernandes, Cláudio e à Prof. Lina Brito pela colaboração nas diversas fases do projecto.

Por último, gostaria também de agradecer a todos os colegas do laboratório de *software 3* de Informática pela ajuda e motivação dada ao longo deste processo.

ABSTRACT

A Wireless Sensor Network (WSN) is composed by a large number of low cost wireless devices (sensors). It uses a set of autonomous sensors that collaborate each other to carry out monitoring of environmental conditions such as temperature, oxygen, light, humidity, pressure, and pollutants, among others. These networks can work for extended periods of time without human intervention, only depending on nodes' battery.

In this context, the WISE-MUSE project (Environmental Monitoring based on Wireless Sensor Networks for the preservation of artworks and historical archives) was created. This project allows implementing museum monitoring solutions and automatic control in real-time.

Currently, the collaboration between the user and the WSN is limited, since sensor nodes have to collaborate each other to transmit its data. Thus, in order to enhance this collaboration, CWSN model (Collaborative Wireless Sensor Network Model) was proposed. CWSN is a collaboration model for WSN, which proposes the use of collaborative sessions in network monitoring.

In order to achieve maximum network performance, it is necessary to define the network deployment. In outdoor locations or in difficult access locations, sensor nodes are placed randomly, for example, launched from an airplane. In indoor places sensor node are placed manually.

In this context, this Master thesis in Computer Sciences has to main contributions: (i) a propagation study in the Museum of the Wale; and; (ii) WISE-MANager, a management system for collaborative sessions.

In order to define the deployment of the network to be installed at the Museum of the Wale, it will present an empirical signal propagation study, which determined the best physical position of the nodes to get a good performance.

The WISE-MANager system allows creating, monitoring and managing collaborative sessions using the Zigbee protocol. The purpose of using collaborative sessions is to provide a better interaction between the user and the WSN, since the user can customize the type of monitoring to be carried out (sensor node, phenomenon or time interval), and interrogate the network and its components. In this way, WISE-MANager increases the flexibility of the WSN.

WSN management is very important, since it allows the user to have a total control of the network devices. In this way, WISE-MANager, allows Museum's managers analyze the network and detect potential problems.

KEY WORDS

ZigBee, XBee, Collaborative Sessions, WSN, Wireless Sensor Network Management.

RESUMO

As redes de sensores sem fios (WSN- *WSN-Wireless Sensor Network*) utilizam um grande número de dispositivos sem fios (sensores), que são de baixo custo e equipados com interfaces *wireless*. Utilizam um conjunto de sensores autónomos que colaboram entre si para efectuar a monitorização das condições ambientais, tais como: temperatura, oxigénio, luz, humidade, pressão, gases poluentes, entre outras. Estas redes podem operar durante largos períodos de tempo, sem intervenção humana, sendo que esse tempo depende do nível de bateria desses nós.

De modo a que os gestores de um Museu possam gerir de forma mais adequada as obras de arte e arquivos históricos, surge o projecto *WISE-MUSE – Environmental Monitoring based on Wireless Sensor Networks*, que permite implementar soluções para a monitorização museológica, com a utilização de redes de sensores sem fios.

Actualmente, a colaboração entre o utilizador e a WSN é muito ténue, sendo que apenas existe colaboração entre os nós sensores. De forma a aumentar esta colaboração, e no âmbito do projecto *WISE-MUSE* surge o *CWSN – Collaborative Wireless Sensor Network Model*, que define um modelo de colaboração na rede de sensores sem fios, permitindo a utilização de sessões colaborativas para a monitorização da rede.

Com o intuito de obter o máximo rendimento da rede, é necessário definir qual o *deployment* a utilizar. O tipo de *deployment* de uma WSN é a forma como os nós são distribuídos pela rede. Em zonas longínquas, ou de difícil acesso, os nós são colocados de forma aleatória, por exemplo, caem de um avião. Nos locais de fácil acesso, podem ser colocados no local exacto.

Portanto, este projecto de Mestrado de Engenharia Informática apresenta duas contribuições principais: (i) um estudo de propagação no Museu da Baleia; e; (ii) o *WISE-MANager*, um sistema de gestão de sessões colaborativas.

De forma a definir qual o *deployment* da rede a instalar no Museu da Baleia, será apresentado um estudo de propagação do sinal empírico, que determinou a melhor posição física dos nós, para que a rede tenha uma boa performance.

O sistema *WISE-MANager* permite a criação, monitorização e gestão de sessões colaborativas numa WSN baseada no protocolo Zigbee. O intuito da utilização de sessões colaborativas é proporcionar uma melhor interacção entre o utilizador e a WSN, visto que o utilizador pode personalizar o tipo de monitorização a efectuar (por nó sensor, por fenómeno ou por intervalo de tempo), e interrogar à rede e aos seus componentes, aumentando assim a flexibilidade da WSN.

A gestão de redes de sensores sem fios é muito importante para que o utilizador tenha controlo sobre a mesma ao saber quais os dispositivos da rede, assim como o seu nível de energia. Por tanto, através de WISE-MANager, os gestores do Museu serão capazes de analisar a rede, detectar eventuais problemas e obter parâmetros específicos.

PALAVRAS CHAVE

ZigBee, XBee, Sessões Colaborativas, WSN, Gestão de Redes de Sensores sem Fios.

TABELA DE CONTEÚDOS

I. Introdução	1
I.1. Apresentação	2
I.2. Motivação	3
I.3. Objectivos	4
I.4. Estrutura da Dissertação	5
II. Estado da Arte	7
II.1. Introdução	8
II.2. Redes de sensores sem fios	9
II.3. Aplicações que utilizam WSN	10
II.4. Dispositivos numa WSN	11
II.5. Protocolos de comunicação nas WSN	12
II.6. ZigBee	14
II.6.1. Características do ZigBee	14
II.6.2. Tipos de Dispositivos	14
II.6.3. Topologias	15
II.6.4. Comunicação ZigBee	18
II.6.5. Endereçamento	18
II.6.6. Transmissão de Dados	19
II.6.7. Camadas de rede do ZigBee	20
II.6.8. Roteamento	21
II.7. Modelo de Colaboração nas redes de sensores sem fios	23
II.7.1. Definições das entidades	23
II.7.2. Hierarquia de colaboração nas redes de sensores sem fios	24
II.7.3. Tipos de Colaboração	24
II.7.4. Papeis dos utilizadores	27
II.8. Sessões Colaborativas	28
II.9. Trabalhos Relacionados	29
II.9.1. Mote View	29
II.9.2. MonSense	29
II.9.3. Tiny DB	30
II.9.4. MANNA – Management Architecture for Wireless Sensor Networks	30
II.9.5. BOSS – Bridge Of SensorS	30
II.9.6. Comparação das diversas ferramentas	30
II.10. Conclusão	33
III. Desenho Arquitectural	35

Tabela de Conteúdos

III.1.	Introdução.....	36
III.2.	Contextualização e domínio da aplicação	37
III.3.	Requisitos.....	38
III.3.1.	Requisitos não funcionais	38
III.3.2.	Requisitos Funcionais.....	39
III.4.	Protótipos Abstractos Canónicos.....	40
III.5.	Diagramas UML.....	43
III.5.1.	Diagrama UML Comportamental: Casos de utilização	43
III.5.2.	Diagrama UML Comportamental: Diagrama de Actividades.....	45
III.5.3.	Diagrama UML de interacção: Diagrama de Sequência	55
	Ao observar a Figura III.18, verifica que esta representa quais as funções e módulos responsáveis pela criação e gestão de sessões colaborativas.	57
III.5.4.	Diagrama UML Comportamental: Diagrama de transição de estados	57
III.6.	Modelo de Dados: Diagrama Entidade-Relacionamento.....	58
III.6.1.	Descrição das entidades.....	60
III.7.	Conclusão.....	62
IV.	Implementação.....	63
IV.1.	Introdução.....	64
IV.2.	Hardware e <i>Software</i> utilizado na Implementação.....	65
IV.2.1.	<i>Software</i>	67
IV.2.2.	<i>API's</i> utilizadas	67
IV.3.	Estudo de Propagação do Sinal	69
IV.3.1.	Descrição do Cenário	69
IV.3.2.	<i>Software</i> utilizado	70
IV.3.3.	Descrição do Estudo de Propagação.....	72
IV.4.	<i>WISE-MANager</i>	73
IV.4.1.	Collaborative Session Management.....	77
IV.4.2.	WSN Management.....	85
IV.5.	Trabalhos Relacionados	92
IV.6.	Conclusão.....	95
V.	Testes e Resultados.....	97
V.1.	Introdução.....	98
V.2.	Ambiente dos Testes.....	99
V.3.	Testes Realizados	100
V.3.1.	Teste 1	100
V.3.2.	Teste 2	102
V.3.3.	Teste 3	104
V.3.4.	Teste 4	105
V.3.5.	Teste 5	106
V.3.6.	Teste 6	107

V.3.7. Teste 7	109
V.3.8. Teste 8	110
V.3.9. Teste 9	112
V.3.10. Teste 10	114
V.4. Conclusão	115
VI. Conclusão	117
VI.1. Introdução	118
VI.2. Principais contribuições	119
VI.3. Trabalhos futuros	120
VI.4. Balanço Final	121
VII. Referências	123
VIII. Anexos	127
VIII.1. Anexo A	128
VIII.1.1. Primeiro Teste	128
VIII.1.2. Segundo teste	128
VIII.1.3. Terceiro teste	128
VIII.1.4. Quarto teste	129
VIII.1.5. Quinto teste	129
VIII.1.6. Sexto teste	129
VIII.1.7. Sétimo teste	129
VIII.1.8. Oitavo teste	130
VIII.1.9. Resultados Obtidos	130
VIII.1.10. Topologia proposta para a rede do Museu da Baleia	136
VIII.2. Anexo B	141

LISTA DE FIGURAS

Figura II.1: Rede com a topologia em estrela.	16
Figura II.2: Rede com a topologia ponto-a-ponto.	16
Figura II.3: Rede com a topologia em árvore.	17
Figura II.4: Topologia árvore numa WSN [17].	17
Figura II.5: Camadas do protocolo de comunicação <i>ZigBee</i> [17].	20
Figura II.6: Diferentes níveis hierárquicos da colaboração na WSN [13].	27
Figura III.1: PAC de criação de sessão colaborativa.	40
Figura III.2: PAC de visualização de sessões activas.	41
Figura III.3: PAC de resumo das sessões	41
Figura III.4: PAC de exportar dados da sessão.	41
Figura III.5: PAC de apagar sessões.	41
Figura III.6: PAC de informações da rede.	41
Figura III.7: PAC de nós da rede.	42
Figura III.8: Diagrama de casos de utilização.	44
Figura III.9: Diagrama de actividade de “Criar Sessão Colaborativa”.	46
Figura III.10: Diagrama de actividade de “Visualizar sessões activas”.	47
Figura III.11: Diagrama de actividade de “Visualizar amostras de dados”	48
Figura III.12: Diagrama de actividade de “Alterar dados das sessões”.	49
Figura III.13: Diagrama de actividade de “Visualizar informações básicas da rede”.	50
Figura III.14: Diagrama de actividade de “Visualizar componentes activos da rede”.	51
Figura III.15: Diagrama de actividade de “Alterar nome dos componentes da rede”.	52
Figura III.16: Diagrama de actividade de “Apagar Sessão”.	53
Figura III.17: Diagrama de actividade de “Exportar histórico das sessões”	54
Figura III.18 Diagrama de Sequência das Sessões Colaborativas.	56
Figura III.19: Diagrama de transição de estados de uma sessão.	57
Figura III.20: Modelo entidade-relação não normalizado.	59
Figura III.21: Modelo de dados entidade-relação normalizado.	60
Figura IV.1: Tipos de <i>XBee</i>	65
Figura IV.2: Configuração do coordenador da rede.	66
Figura IV.3: Configuração do router utilizado nos testes.	66
Figura IV.4: Escolher porta do coordenador no programa <i>RSSI Level</i>	70
Figura IV.5: Escolher o endereço para saber o nível de sinal.	71
Figura IV.6: Sinal <i>RSSI</i> do endereço <i>00 13 a2 00 40 2d d7 ca</i>	71
Figura IV.7: Criar a conexão entre o <i>XBee</i> e a plataforma.	71
Figura IV.8: Criação do comando DB.	72
Figura IV.9: Envio do comando DB.	72
Figura IV.10: Medições no piso 0.	72
Figura IV.11: Diagrama de classes do sistema <i>WISE-MANager</i>	74
Figura IV.12: Interface inicial da aplicação <i>WISE-MANager</i>	77

Lista de Figuras

Figura IV.13: Escolher os nós para a sessão.....	78
Figura IV.14: Interface de “criação de sessão colaborativa”	79
Figura IV.15: Janela da ajuda na escolha dos nós da sessão.	80
Figura IV.16: Listagem de todas as sessões activas.	80
Figura IV.17: Dados monitorizados pela sessão Sala Cetáceos.	81
Figura IV.18: Nós da sessão “Sala Cetáceos Manhã”	82
Figura IV.19: Listagem das sessões colaborativas.	83
Figura IV.20: Interface da criação de um histórico das sessões.	84
Figura IV.21: Interface de apagar sessões.	85
Figura IV.22: Mensagem de informação da quantidade de amostras guardadas.	85
Figura IV.23: Efectuar a conexão com a rede XBee.	86
Figura IV.24: Criação do comando “OP”	87
Figura IV.25: Envio e recepção do comando “OP”	87
Figura IV.26: Criação do comando “CH”	87
Figura IV.27: Envio e recepção do comando “CH”	87
Figura IV.28: Envio do comando NC para todos os dispositivos da rede.	88
Figura IV.29: Envio do comando NI para o dispositivo.	89
Figura IV.30: Envio do comando “ND”	89
Figura IV.31: <i>Packet Listener</i> para receber os pacotes da rede.	89
Figura IV.32: Informações da rede.	90
Figura IV.33: Dispositivos da rede.	91
Figura V.1: Coordenador da rede.	99
Figura V.2: Router.	99
Figura V.3: Nó sensor.	99
Figura V.4: Esquema de montagem da rede no Teste 1.	100
Figura V.5: Criação da sessão colaborativa utilizada no teste 1.	101
Figura V.6: Resultados obtidos com a sessão Teste 1.	102
Figura V.7: Esquema de montagem da rede no Teste 2.	102
Figura V.8: Sessões activas no Teste 2.	103
Figura V.9: Resultados da monitorização da Sessão: Teste 2 – Sessao 1.	103
Figura V.10: Resultados da monitorização da Sessão: Teste 2 – Sessao 2.	104
Figura V.11: Esquema de montagem da rede no Teste 3.	104
Figura V.12: Resultados da monitorização utilizando a sessão Teste 3.	105
Figura V.13: Esquema da rede utilizado no Teste 4.	105
Figura V.14: Dispositivos da rede no Teste 4.	106
Figura V.15: Resultados da sessão Teste 4.	106
Figura V.16: Resultados da sessão Teste 5.	107
Figura V.17: Esquema da rede utilizado no Teste 6.	107
Figura V.18: Sessão: Teste 6 – Sessão 1.	108
Figura V.19: Sessão: Teste 6 – Sessão 2.	108
Figura V.20: Resultados da sessão Teste 6 – Sessao 1.	108
Figura V.21: Resultados da sessão Teste 6 – Sessao 2.	109
Figura V.22: Esquema da rede utilizado no Teste 7.	109
Figura V.23: Resultado da sessão Teste 7 – Sessão 1.	110
Figura V.24: Resultado da sessão Teste 7 – Sessão 2.	110
Figura V.25: Esquema de montagem da rede.	111
Figura V.26: Informações da rede de sensores sem fios.	111

Figura V.27: Dispositivos da rede de sensores sem fios.	112
Figura V.28: Resultados da sessão Divisão A.....	113
Figura V.29: Resultados da sessão Divisão B.	113
Figura V.30: Esquema da rede no teste 10.	114
Figura V.31: Resultados obtidos no teste 10.	114
Figura VIII.1: Medições no piso 0.....	130
Figura VIII.2: Medições no piso 0.....	131
Figura VIII.3: Valores medidos no piso 1.....	132
Figura VIII.4: Valores medidos no piso 2.....	133
Figura VIII.5: Medições realizadas no piso -1.	134
Figura VIII.6 – Medições realizadas no piso -1.	135
Figura VIII.7: Medições no piso 0, na zona da sala do segurança.....	135
Figura VIII.8: Medições no piso 0, da sala do segurança até a sala do cofre.	136
Figura VIII.9: Localização dos nós, bloqueadores e routers no piso 0.....	137
Figura VIII.10: Localização dos nós, bloqueadores e routers no piso 1.....	138
Figura VIII.11: Localização dos nós, bloqueadores e routers no piso 2.....	139
Figura VIII.12: Localização dos nós, bloqueadores e routers no piso -1.	140
Figura VIII.13: Excerto do ficheiro gerado do histórico de uma sessão.	141

LISTA DE TABELAS

Tabela II.1: Comparação entre as tecnologias <i>Bluetooth</i> , <i>ZigBee</i> e <i>IEEE 802.11b</i> ..	13
Tabela II.2: Tradução endereços de 64 bits-16 bits.	20
Tabela II.3: Comparação em relação aos parâmetros de visualização.....	31
Tabela II.4: Comparação em relação aos parâmetros de gestão de rede.....	31
Tabela II.5: Escolher o que monitorizar na WSN.....	32
Tabela II.6: Interrogar a rede.....	32
Tabela IV.1: Características dos <i>XBee</i>	65
Tabela IV.2: Estados das portas de emergência.	69
Tabela IV.3: Comparação dos parâmetros de visualização entre as ferramentas e o sistema <i>WISE-MANager</i>	92
Tabela IV.4: Comparação dos parâmetros de gestão entre as ferramentas e o sistema <i>WISE-MANager</i>	92
Tabela IV.5: Escolher o que monitorizar na WSN.	92
Tabela IV.6: Interrogar a rede.	93
Tabela VIII.1 – Valores medidos nos pontos A e B na sala de cetáceos.	131
Tabela VIII.2 – Testes realizados para determinar a localização do <i>router</i> R2...	131
Tabela VIII.3: Valores obtidos entre os routers.	132
Tabela VIII.4 – Valores medidos para obter a melhor localização de R5.	133
Tabela VIII.5 – Valores medidos no piso -1.	134

ACRÓNIMOS

API – Application Programming Interface

AC – Apply Changes

ASCII – American Standard Code for Information Interchange

AT – Transparent Mode

CH – Operating Channel

CO – Monóxido de Carbono

CO2 – Dióxido de Carbono

ID – Identifier

IDE – Integrated Development Environment

IEEE – Institute of Electrical and Electronics Engineers

MAC – Media Access Control

MS – Microsoft

NC – Number of Children

ND – Node Discovery

NI – Node Identifier

OP – Operating PAN ID

OSI – Open Systems Interconnection

PAN – Personal Area Network

PAN ID – Personal Area Network Identifier

PDA – Personal Digital Assistant

RF – Radio-Frequência

RSSI – Received Signal Strength Indication

SQL – Structured Query Language

SWT – Standard Widget Toolkit

WISE-MUSE – Wireless Sensor Museum Enviroment

WR – Write

WSN – Redes de Sensores Sem fios

WSN – Wireless Sensor Network

WSNSCW – Wireless Sensor Network Supported Cooperative Work

I. INTRODUÇÃO

I.1. APRESENTAÇÃO

Uma *WSN - Wireless Sensor Network* - é uma rede que utiliza um conjunto de sensores autónomos que colaboram entre si para efectuar a monitorização das condições ambientais, tais como, temperatura, oxigénio, luminosidade, humidade, pressão, gases poluentes, entre outras. Este tipo de rede utiliza um grande número de dispositivos sem fios (sensores), de baixo custo e equipados com interfaces *wireless*, facilitando a comunicação. Estes sensores são capazes de efectuar processamento, de recolher informações de outros nós da rede e de comunicar com esses nós. As redes de sensores sem fio podem operar durante largos períodos sem a intervenção humana, sendo que esse tempo depende da quantidade de energia de cada sensor [1].

Através da gestão das redes de sensores sem fios pode-se visualizar quais os componentes e qual a possível escalabilidade da rede. Permite também a visualização de problemas na rede, dando a possibilidade ao gestor da rede resolvê-los remotamente.

No âmbito do projecto *WISE-MUSE*, que permite efectuar a monitorização ambiental de museus utilizando redes de sensores sem fios para a conservação de obras de arte, encontra-se este projecto de Mestrado em Engenharia Informática, que visa o desenvolvimento de um sistema que seja capaz de efectuar a gestão da rede de sensores sem fios, criando e utilizando sessões colaborativas, além do envio e recepção de comandos da rede. Este sistema permite ao utilizador ter uma relação mais estreita e de maior controlo sobre a *WSN*, controlando a sua execução e monitorização. Além disto, o utilizador tem possibilidade de efectuar a gestão da rede, visualizando o estado e controlando os dispositivos da mesma.

I.2. MOTIVAÇÃO

As *WSN's* são por natureza redes colaborativas, visto que necessitam de colaborar para conseguirem completar as suas tarefas, sejam elas saber os valores do ambiente, processamento de sinal, computação, encaminhamento, localização, segurança, entre outras. A colaboração existente neste tipo de redes é muito ténue, sendo que apenas existe entre os dispositivos da rede, e não entre o utilizador e a *WSN*, que apenas observa as amostras recebidas.

Com a falta deste tipo de colaboração, surgiu o modelo *CWSN – Collaborative Wireless Sensor Network*- que permite modelar toda a *WSN*, mostrando quais as entidades, propriedades, relações e estados, apresentando como pode ser efectuada a colaboração entre o utilizador e a rede para alcançar um objectivo comum, ou seja, a monitorização dos fenómenos ambientais [3].

A criação de sessões colaborativas permite aumentar a flexibilidade da rede, de tal forma que o utilizador pode controlar directamente a mesma, personalizando a actividade de monitorização, definindo dentro das sessões o tempo de monitorização, quais nós sensores actuam nela e que parâmetros ambientais monitorizam. Actualmente, existem várias ferramentas de gestão de rede de sensores. No entanto, nenhuma delas permite a criação e gestão de sessões colaborativas.

Devido a isto, surgiu a necessidade do desenvolvimento de uma ferramenta gráfica de gestão e criação de sessões colaborativas, baseado no modelo *CSWN*, que, activando e desactivando as sessões, permita uma poupança de energia, em certos intervalos de tempo. Com a utilização de sessões colaborativas, a visualização das amostras da rede será totalmente personalizável por parte do utilizador, tendo em conta os parâmetros (quais os nós a monitorizar, quando estes monitorizam, e o que monitorizam) utilizados na criação da mesma.

I.3. OBJECTIVOS

Neste projecto de mestrado pretendeu-se efectuar a gestão de uma rede de sensores sem fios utilizando sessões colaborativas, assim como efectuar um estudo de propagação do sinal no Museu da Baleia. De forma mais específica, os objectivos são os seguintes:

- Efectuar um estudo teórico sobre a gestão de redes de sensores sem fios.
- Efectuar o estudo de propagação do sinal no Museu da Baleia para obter o melhor *deployment* da WSN.
- Desenvolver uma ferramenta de gestão da rede que permita efectuar a criação e gestão de sessões colaborativas e gestão de redes baseadas no protocolo *ZigBee*.
- Permitir que seja efectuada a gestão de uma WSN remotamente, através do envio e recepção de comandos.
- Utilizar os nós sensores para formar sessões colaborativas, que terá o intuito de monitorizar o meio ambiente (humidade, luminosidade, temperatura, monóxido e dióxido de carbono) das salas do Museu da Baleia.
- Permitir a integração com a plataforma *WISE-MUSE* já existente.

I.4. ESTRUTURA DA DISSERTAÇÃO

Esta tese de Mestrado está dividida em seis capítulos. No primeiro capítulo é efectuada a introdução, onde apresentamos o tema do projecto, assim como quais as motivações e objectivos do mesmo. De seguida, no capítulo dois é efectuado um enquadramento com os trabalhos relacionados, assim como o estado da arte das tecnologias a utilizar. No terceiro capítulo, é descrito o desenho do sistema, apresentando diversos diagramas *UML* assim como o modelo de dados a utilizar. No capítulo seguinte, é mostrado como foi implementado o sistema descrito no capítulo três, apresentando o sistema e as suas características e capacidades. No capítulo cinco, encontra-se a validação do sistema, através dos testes efectuados ao mesmo. Finalmente, no capítulo seis, encontra-se a conclusão da tese, assim como quais as contribuições efectuadas por este projecto de mestrado e também quais os trabalhos futuros.

II. ESTADO DA ARTE

II.1. INTRODUÇÃO

Neste capítulo será abordado o estado da arte das redes de sensores sem fios, mais concretamente, da tecnologia a utilizar, o *ZigBee*. Também será abordado o estado da arte de algumas ferramentas de monitorização e gestão de redes de sensores sem fios. Irá ser abordado igualmente o modelo a partir do qual efectuou-se as sessões colaborativas, o *“Hierarchical Modelling of Collaboration in Wireless Sensor Networks”*.

II.2. REDES DE SENSORES SEM FIOS

Uma WSN utiliza um grande número de sensores para efectuar a monitorização do ambiente. Estes sensores são dispositivos de baixo custo, equipados com interfaces *wireless*, o que permite enviar os dados para a internet, por exemplo [1,4].

As redes de sensores sem fio possuem uma autonomia que permite actuar durante largos períodos de tempo.

Uma rede de sensores deverá utilizar endereçamento baseado em atributos, podendo desta forma os sensores responder a determinadas situações que ocorram na rede, por exemplo, temperatura maior que 30°, mostrando os sensores que estiverem nessa situação. Os sensores devem conhecer os seus nós vizinhos, de forma a facilitar o envio de dados para os mesmos.

As redes de sensores sem fios têm a vantagem de poderem ser utilizadas em ambientes difíceis, visto que a mesma tem capacidade de lidar com falhas nos nós. Este tipo de rede permite a mobilidade dos nós e também proporciona uma topologia dinâmica da rede. Neste tipo de rede podemos ter um grande número de dispositivos, sendo que é muito fácil aumentar a rede, visto que o limite da rede é apenas a largura de banda do nó *router*.

II.3. APLICAÇÕES QUE UTILIZAM WSN

As redes de sensores sem fios começaram a ser implementadas em 1990 na engenharia do ambiente, agricultura e em aplicações ecológicas [6].

As aplicações mais comuns deste tipo de rede envolvem monitorização e controlo. Pode ser utilizado para monitorizar variáveis ambientais em locais internos como prédios, ou em locais externos, como florestas ou desertos. É utilizado também na monitorização do tráfego, na segurança de edifícios ou na medicina, em que pode monitorizar o funcionamento de órgãos, ou detectar a presença de substâncias que indicam algum problema biológico [1, 2, 10].

Outra das áreas de aplicação das WSN é no ramo do petróleo e gás. A utilização deste tipo de rede diminui os custos da procura de novos poços de petróleo e gás para explorar. Existem já muitas redes de sensores sem fios a serem utilizadas com este objectivo, e muitas delas têm mais de 500 nós [7].

Outras das utilizações deste tipo de rede são nas aplicações domésticas, onde poderá ser utilizado automatização doméstica, como por exemplo sensores embutidos em electrodomésticos.

II.4. DISPOSITIVOS NUMA WSN

A rede deve permitir que os utilizadores da mesma comuniquem com os nós através de uma estação base, conhecida por *sink*. Estas comunicações podem ser utilizadas para enviar alguma informação para os sensores, ou para retirar alguma informação dos sensores [4].

Neste tipo de rede temos vários dispositivos [10]:

- *Sensor nodes*: São os dispositivos autónomos com capacidade de monitorização ambiental, processamento e comunicação. Quando estes nós estão numa rede, formam uma rede de sensores. Recolhem os dados através dos sensores, processam os dados localmente, e depois enviam os dados até à estação base, utilizando outros nós, numa transmissão *multi-hop*, para passar a informação até ao nó âncora ou estação base.
- *Anchor node*: Nó que difere dos outros tipos de dispositivos visto que sabe a sua localização através de um receptor *GPS*. O seu objectivo é ajudar os outros nós no processo de localização. Desta forma, todos os nós conseguem saber onde se encontram.
- *Cluster Head*: Definido aquando da aplicação de um algoritmo de *clustering*. É o nó que efectua a comunicação entre os sensores desse cluster e a estação base.
- *Sink Node*: Dispositivo que efectua a comunicação entre o utilizador e os sensores. É através dele que os dados dos sensores chegam ao utilizador.

II.5. PROTOCOLOS DE COMUNICAÇÃO NAS WSN

Foram propostos alguns padrões para utilizar na interligação dos sensores nas WSN, entre eles o padrão para redes locais *wireless IEEE 802.11*, o padrão para redes residenciais *HomeRF* e o padrão para interconexão de dispositivos *Bluetooth* [10].

O padrão *IEEE 802.11* permite criar uma *Wireless LAN (WLAN)*, ou seja, cria um sistema de comunicação de dados flexíveis e quando necessário, é possível que computadores portáteis ou *PDA*s liguem-se de forma rápida a uma *WLAN*. Este protocolo foi proposto visando estabelecer um padrão para redes locais sem fio para comunicação de dados com taxas de transferência de até 2 Mbits/s. A primeira versão do padrão foi publicada em 1997, e possibilitava o uso de rádio frequência e infravermelho para a comunicação. Em 1999, foram publicadas duas versões padrão pelo IEEE, o *802.11a* e o *802.11b HR*, que possibilitava taxas de transferência de até 54 Mbits/s e 11 Mbits/s, respectivamente. Estas novas versões utilizavam apenas rádio frequência, sendo que esta tecnologia estava direccionada para interligação de diferentes tipos de dispositivos computacionais [10].

O padrão *HomeRF* foi criado e desenvolvido em 1998 pelo *HomeRF Working Group*, um grupo que inclui a *Siemens*, *Motorola* e a *Philips*. Este padrão visa interligar equipamentos digitais numa rede local sem fio. Muito semelhante ao padrão *IEEE 802.11*, o *HomeRF* adiciona tráfego de voz no seu protocolo. Actualmente, proporciona taxas de transferência até 10 Mbits/s. Este padrão permite aceder a *access points* a 50 metros de distância de modo a garantir que o dispositivo permanece na mesma *PAN (Personal Area Network)*.

Outra das opções é a tecnologia *bluetooth*, uma especificação para áreas de redes pessoais sem fio, *PANs*. Esta tecnologia permite conectar e trocar informações entre dispositivos que possuam a tecnologia *bluetooth*, ou que tenham uma interface *bluetooth*. Para utilizar esta tecnologia, seria necessário equipar todos os intervenientes na comunicação com adaptadores *bluetooth*. A taxa de transferência dos vários dispositivos *bluetooth* é de sensivelmente 1 Mbit/s. O alcance do sinal é de aproximadamente 10 metros, podendo chegar aos 100 metros, dependendo do dispositivo.

O protocolo *ZigBee* foi criado de forma a interligar pequenas unidades de recolha de dados, como por exemplo os sensores das WSN. Esta tecnologia tem a vantagem, em relação às anteriores, de garantir um menor consumo com alcance reduzido (cerca de 10 metros), garantindo que a comunicação entre duas unidades pode ser repetida sucessivamente pelas unidades existentes na rede até chegar ao destino final. Todos os pontos da rede funcionam como repetidores, retransmitindo a informação numa malha de unidades *ZigBee* [5].

As soluções *ZigBee* são mais apropriadas para aplicações que envolvam dispositivos remotos alimentados por baterias, nomeadamente sensores, já que permite baixos consumos, o que leva a que as baterias não necessitem de ser trocadas tão frequentemente [8].

O *ZigBee* não é um concorrente directo do *bluetooth*, visto que o *ZigBee* foi desenvolvido para realizar monitorização de sistemas e o *bluetooth* é mais apropriado para aplicações que necessitam de um maior débito, como por exemplo as redes *Ad-hoc*, como podemos observar pela comparação na Tabela II.1:

Tabela II.1: Comparação entre as tecnologias *Bluetooth*, *ZigBee* e *IEEE 802.11b*

	Taxa de transmissão	Alcance (<i>indoor</i>)
<i>ZigBee</i>	20 a 250 Kbps	10-100 m
<i>Blueetooth</i>	1 a 3 Mbps	2-10 m
<i>IEEE 802.11b</i>	1 a 54 Mbps	30-100 m

Observando a Tabela II.1, verificamos que existem algumas diferenças entre as duas tecnologias. A capacidade de débito do *Bluetooth* é muito maior, mas para as redes de sensores sem fios não é necessária uma capacidade de transferência de dados tão alta. As grandes vantagens da tecnologia *ZigBee*, estão na duração das baterias e na quantidade de dispositivos que é permitido estar na mesma rede.

II.6. ZIGBEE

II.6.1.Características do ZigBee

ZigBee é um standard que define um conjunto de protocolos de comunicação para dispositivos, com baixas taxas de transmissão de dados, sem fios. Os dispositivos *ZigBee* operam nas bandas de frequências 868 MHz (que é utilizada na Europa por várias aplicações), 915 MHz e 2.4 GHz, sendo que estas duas bandas são utilizadas maioritariamente na área industrial, científica e médica. A frequência 915 MHz é utilizada maioritariamente na América do Norte enquanto que a banda 2.4 GHz é utilizada em toda a parte. A escolha geralmente recai pela frequência de 2.4 GHz visto que é a que permite uma taxa de transmissão mais elevada e também disponibiliza um maior número de canais [16].

A taxa de transmissão máxima destes dispositivos é de 250 kbps. Estes tipos de dispositivos são muito utilizados em redes onde não são necessárias grandes taxas de transmissão de dados e também onde é necessário que os custos não sejam elevados. O alcance máximo de transmissão *indoor* é de 150 metros e permite um máximo de 65535 nós por coordenador [16].

A especificação *ZigBee* define as camadas de rede, aplicação e o serviço de segurança. A definição das camadas física e de acesso ao meio (MAC) é da responsabilidade da norma *IEEE 802.15.4*. Complementando esta norma, no que respeita às restantes camadas do modelo *OSI*, o *ZigBee* tem como principais características o facto de ser um protocolo para aplicações de baixo consumo e de baixo custo. Estas duas características tornam-no ideal para sistemas de monitorização.

O protocolo *ZigBee* é o futuro das redes de sensores sem fios, visto que, com uma implementação bastante simples garante um consumo muito baixo. Permite também um baixo custo de instalação e manutenção dos dispositivos da rede, possibilitando também a utilização de uma grande densidade de nós [8].

II.6.2.Tipos de Dispositivos

O protocolo *ZigBee* define três tipos de dispositivos: coordenador, *router* e *end-device*:

O coordenador é um dispositivo indispensável na rede visto que é o único capaz de iniciar a rede e efectuar a manutenção da mesma. Quando cria a rede, tem que seleccionar um bom canal para a rede operar. Para isto, efectua um *scan* em vários canais (frequências) para verificar os níveis de energia em todos eles. Os canais que contenham níveis de energia muito altos são retirados da lista de potenciais canais escolhidos. Ao seleccionar o canal, escolhe um *PAN ID* que não

esteja em utilização, começando assim a difundir a rede. Feito isto, permite que os *routers* e os *end-devices* se conectem a essa mesma rede. Também permite o encaminhamento de pacotes e a comunicação com outros dispositivos na rede. Todas as redes *ZigBee* têm que ter um, e só um, coordenador, independentemente da topologia da rede, e este nunca poderá estar no modo *sleep*, visto que se isso acontecesse, não seria capaz de receber os dados dos outros dispositivos, nem de difundir a rede.

Outro dos dispositivos da rede é o *router*, que funciona como intermediário, permitindo a comunicação entre nós sem intervenção do coordenador. Tem que se juntar a uma *PAN* – *Personal Area Network* – antes de conseguir transmitir, receber ou encaminhar dados. Depois de se juntar à rede, pode possibilitar a entrada de outros *routers* e *end-devices* na rede. Este dispositivo não poderá estar no estado *sleep*, visto que isso faria com que os pacotes enviados pelos *end-devices* se perdessem. A principal função dos *routers* é a retransmissão de mensagens de um nó para outro.

Finalmente, o outro dispositivo da rede é o *end-device*. Tal como os anteriores, para transmitir ou receber dados têm que se juntar a uma *PAN*. Este tipo de dispositivo não permite que se juntem novos dispositivos à rede e também não efectua o encaminhamento dos dados. Efectua apenas a monitorização da rede, enviando esses dados para o coordenador ou *router*. Pode entrar em modos que permitam conservar energia, no modo *sleep* por exemplo.

O coordenador e os *routers* possibilitam entradas de novos dispositivos na rede, e, ao acontecer isto, ficam como *parent* desse dispositivo, levando a que todas as comunicações para/de o dispositivo tenham que passar por si. O *parent* também actua como buffer, guardando os pacotes destinados ao seu filho, até que este saia do estado *sleep*.

II.6.3.Topologias

A formação da rede é controlada pela camada de rede do *ZigBee*, sendo que utiliza sempre uma das duas topologias especificadas pelo *IEEE 802.15.4*: em estrela ou ponto-a-ponto. A topologia ponto-a-ponto contém as derivações em malha e em árvore.

A topologia em estrela, na Figura II.1, é a mais simples e a mais limitada das topologias que utilizam a tecnologia *ZigBee*. Uma rede em estrela consiste num coordenador e num conjunto de *end-devices*, sendo que cada *end-device* só pode comunicar com o coordenador, ficando no perímetro da rede. Desta forma, para um *end-device* mandar mensagem para outro, tem que comunicar com o coordenador, que encaminha a mensagem até o destino. Poderão ser utilizados *routers* no lugar dos *end-devices*, mas estes não iriam utilizar a função de encaminhamento de pacotes.

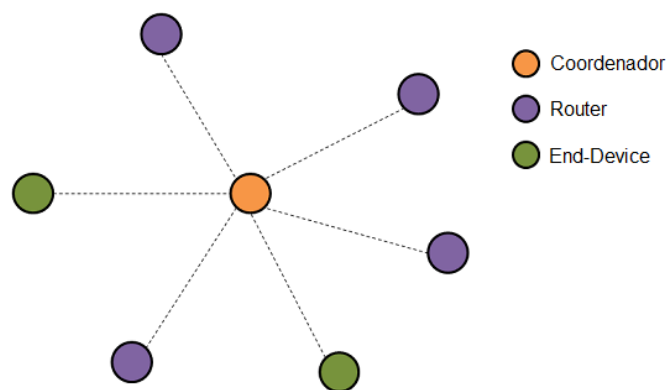


Figura II.1: Rede com a topologia em estrela.

Como podemos observar na Figura II.1, uma das desvantagens desta topologia é a não existência de um caminho alternativo caso a comunicação entre o coordenador e o *end-device* falhe. Outra desvantagem será o congestionamento que poderá ser causado no coordenador devido à constante comunicação com os *end-devices*. Para utilizar este tipo de topologia, é necessário programar um dispositivo para ser o coordenador da rede, que uma vez activado, cria a rede. Ao criar a rede, selecciona um identificador de rede único (*PAN ID*), ou seja, que não é utilizado por nenhuma outra rede, garantindo assim que não existem interferências com outras redes criadas anteriormente. Depois disto, é necessário configurar o *PAN ID* nos outros dispositivos que se irão juntar à rede. [16].

A rede ponto-a-ponto, na Figura II.2, é uma rede que garante um grande nível de fiabilidade e escalabilidade, visto que fornece mais de um caminho ao longo da mesma. Neste tipo de rede, os dispositivos comunicam directamente com qualquer outro dispositivo, desde que estejam colocados perto o suficiente. O coordenador e o *router* podem encaminhar pacotes, permitindo assim uma maior flexibilidade na propagação das mensagens, visto que existem rotas alternativas, logo, se um caminho falhar não existe congestionamento da rede. É utilizada uma função de descoberta de rotas de forma a fornecer o melhor caminho disponível para efectuar a comunicação [9].

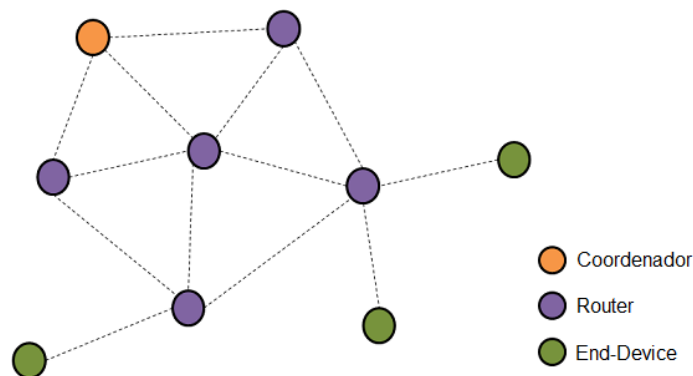


Figura II.2: Rede com a topologia ponto-a-ponto.

Uma rede ponto-a-ponto pode tomar várias formas quando são criadas restrições na comunicação dos dispositivos. Isto é efectuado com a colocação de *end-devices*, que não têm a capacidade de encaminhar pacotes, ficando desta forma apenas conectados directamente ao *router* que foi responsável pela sua entrada na rede.

Outra topologia que é utilizada neste tipo de rede é a topologia em árvore, onde temos o coordenador que estabelece a rede inicial e permite a conexão de vários *routers*, que formam os ramos e encaminham as mensagens. Os *end-devices* funcionam como folhas, não participando no encaminhamento dos pacotes. A Figura II.3 mostra um exemplo deste tipo de rede.

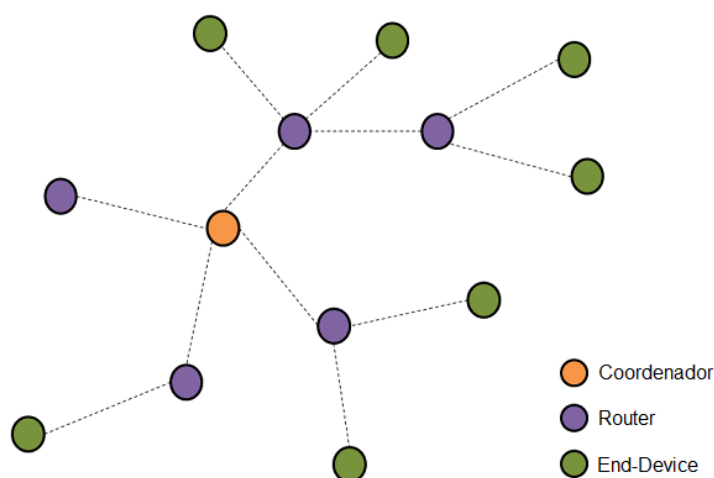


Figura II.3: Rede com a topologia em árvore.

Na Figura II.4 podemos observar uma das vantagens da topologia em árvore. Este tipo de topologia permite o contorno de obstáculos, utilizando os diversos *routers* da rede para encaminhar os dados. Existe comunicação entre os *routers* e o coordenador, sendo que os *end-devices* apenas podem comunicar com um *router* (nó pai), que em seguida encaminha a sua mensagem até o destino.

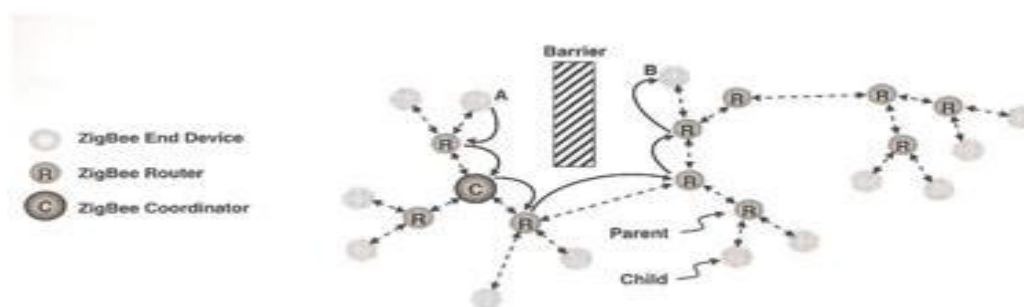


Figura II.4: Topologia árvore numa WSN [17].

As redes baseadas em *cluster-tree*, utilizam uma topologia baseada na rede em estrela/malha, combinando os benefícios das duas, para altos níveis de fiabilidade e utilização de nós com bateria. Este tipo de rede consiste em várias estrelas directamente conectadas entre si, onde os nós centrais estão directamente em

comunicação com o único coordenador. Utilizando um conjunto de *routers* e um único coordenador, a rede fica interconectada com uma malha de *routers* e *end-devices* que passam a informação de nó a nó utilizando o melhor caminho possível. Se algum *router* ficar indisponível, os outros *routers* irão descobrir novas rotas para nenhum nó ficar incontactável. Assim, garantimos que ficamos com uma rede robusta e fiável [17].

Concluindo, independentemente da topologia, uma rede *ZigBee* é sempre criada através de um coordenador *PAN*, que controla a rede e efectua:

- Distribui o endereço de rede único (16-bit) a todos os dispositivos que se conectam à rede;
- Inicia, termina e encaminha as mensagens pela rede;
- Selecciona o identificador da rede (*PAN ID*), que permite que os dispositivos dentro da rede utilizem apenas o endereço de 16-bits para comunicarem.

O coordenador (ao contrário dos *end-devices*) necessita de estar sempre ligado, e devido a isso necessita de estar conectado à rede eléctrica. A rede mais pequena que pode ser criada é composta por um coordenador e um outro dispositivo, que pode ser um *router* ou um *end-device*.

II.6.4.Comunicação *ZigBee*

Existem três tipos de transferência de dados na comunicação *ZigBee*:

- Do coordenador para o dispositivo;
- Do dispositivo para o coordenador;
- Entre dois dispositivos.

Estes três métodos são utilizados na topologia ponto-a-ponto. Na topologia em estrela, apenas existem os dois primeiros tipos de comunicação. Na rede ponto-a-ponto, com a topologia em árvore, existem três tipos de dispositivos: coordenador, *router* e *end-device*. A comunicação efectua-se entre o coordenador e o *router*, e vice-versa, e também entre coordenador/*router* e o *end-device*. Não existe comunicação entre os *end-devices* [16].

II.6.5.Endereçamento

Todos os dispositivos *ZigBee* têm dois tipos de endereços diferentes, um que é o endereço *IEEE*, o identificador único de cada dispositivo, e outro que é o endereço de rede, que é atribuído quando um dispositivo se junta a uma rede *ZigBee* [12]:

O endereço *IEEE*, o endereço de 64 bits, é o endereço que identifica o dispositivo garantindo que não existem dois dispositivos com o mesmo endereço a nível físico.

O endereço de rede, o endereço de 16 bits, identifica o nó na rede. Este endereço é local para aquela rede, logo podem existir dois dispositivos em redes diferentes com o mesmo endereço de 16 bits. Os endereços de rede são alocados pelo nó pai (*router* ou coordenador) quando o nó filho se junta à rede. O endereço de rede do coordenador está predefinido e é o *0x0000*, sendo que para os outros dispositivos da rede, o endereço pode ser qualquer um que ainda se encontre livre. Na Tabela II.2 estão apresentados dois endereços de dispositivos que utilizam o protocolo *ZigBee*, o endereço de 64-bits e o de 16-bits.

Todas as transmissões na rede são feitas utilizando o endereço de rede dos dispositivos de origem e destino. A utilização deste endereço para a comunicação permite uma redução no tamanho das mensagens poupando desta forma memória aquando da gravação dos endereços da rede. A utilização de um *PAN ID* único e do endereço de rede faz com que a comunicação entre as várias redes seja independente.

As tabelas de endereçamento dos dispositivos *ZigBee* também utilizam os endereços de 16-bit para determinar como efectuar o encaminhamento dos pacotes ao longo da rede. No caso de haver um reset na rede, os endereços de 16 bits dos dispositivos *ZigBee* podem ser alterados. Devido a isto, todas as transmissões incluem o endereço de 64 bits dos dispositivos, para garantir que a transmissão chega ao destino.

II.6.6. Transmissão de Dados

Os pacotes nas redes *ZigBee* podem ser transmitidos em *unicast* ou *broadcast*. As transmissões *unicast* acontecem quando existe comunicação de um dispositivo de origem para um dispositivo de destino, enquanto que as transmissões *broadcast* são efectuadas para muitos ou para todos os nós da rede [12]:

- As transmissões *broadcast* têm o objectivo de propagar o sinal ao longo de toda a rede para que todos os nós recebam a transmissão.
- As transmissões *unicast* são efectuadas de um nó para o outro. As transmissões podem ser efectuadas para um nó vizinho ou para um nó que esteja a vários saltos de distância.

Todos os dispositivos *ZigBee* têm uma tabela de endereços que contém os endereços de 64 bits e 16 bits, como podemos ver na Tabela II.2. Quando é efectuada uma transmissão para o endereço de 64 bits, o dispositivo verifica na tabela de endereços se tem a correspondência para o endereço de 16 bits. Se não houver correspondência, é lançado um algoritmo de descoberta de endereços para descobrir o endereço de 16 bits correspondente.

Tabela II.2: Tradução endereços de 64 bits-16 bits.

Endereço de 64 bits	Endereço de 16 bits
0x0013a200402dd7ca	0x4c3d
0x0013a200403c4c76	0x0000

Existem dois modos de transmissão de pacotes nas redes *ZigBee*, o modo *AT* e o modo *API*. Se o objectivo for criar uma rede simples ponto-a-ponto entre dois dispositivos, podemos utilizar o modo *AT*, onde existe uma comunicação serial entre os dispositivos. O modo *API* oferece maiores capacidades de configuração dos vários dispositivos da rede, permitindo a comunicação com cada um dos dispositivos individualmente, assim como uma comunicação *broadcast* para todos os dispositivos da rede.

II.6.7.Camadas de rede do *ZigBee*

O *ZigBee* consiste num conjunto de camadas que comunicam entre si para possibilitar o correcto funcionamento do protocolo. Um resumo dessas camadas está na Figura II.5. De seguida estão apresentadas todas as camadas [11,16]:

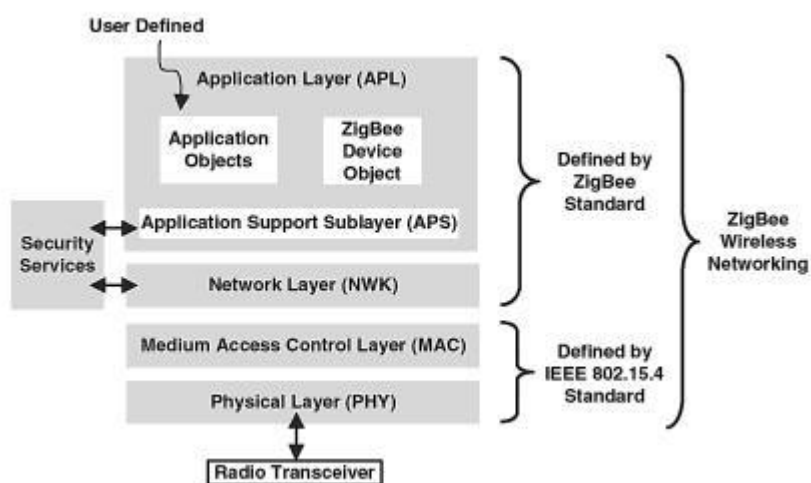


Figura II.5: Camadas do protocolo de comunicação *ZigBee* [17].

Nesta Figura podemos visualizar que existem 5 camadas neste protocolo. Duas são definidas pelo standard *IEEE 802.15.4* e as outras três são definidas pelo standard *ZigBee*.

II.6.7.1.Camada Física

A camada física (*Physical Layer*) é a camada mais próxima do hardware, controlando e comunicando directamente com o dispositivo *ZigBee*. Esta camada é responsável por activar o rádio que transmite ou recebe pacotes. Esta camada

também selecciona a frequência utilizada e garante que o canal utilizado não está em uso por outros dispositivos, nem redes. As comunicações são feitas através de pacotes, sendo constituído por três partes: o *header* de sincronização (SHR), o *header* físico (PHR) e o *payload*. O SHR – *Synchronization Header* – permite que o receptor sincronize-se de forma a receber o pacote. O PHR – *PHY header* – contém o tamanho do pacote enquanto que o *payload* é fornecido pelas camadas superiores, e contém informações como dados ou comandos para o dispositivo receptor [16].

II.6.7.2.Camada MAC

A camada seguinte é a MAC – *Medium Access Control* – que fornece a interface entre as camadas físicas e de rede. É a camada responsável por efectuar a sincronização entre os dispositivos, fornecendo também os serviços de associação e dissociação [16].

II.6.7.3.Camada de Rede

A camada de rede (*Network Layer*) efectua a interface entre as camadas MAC e de aplicação. Esta camada também é responsável pela criação da rede e do encaminhamento de pacotes na mesma. Desta forma, a camada de rede do coordenador *ZigBee* é responsável pela criação da nova rede e de seleccionar a topologia da rede, atribuindo os endereços de rede aos dispositivos da mesma. Com a criação das rotas, os pacotes de dados são transmitidos utilizando vários dispositivos desde o dispositivo de emissão ao receptor [16].

II.6.7.4.Camada de Aplicação

A camada de aplicação (*Application Layer*) é a camada de mais alto nível na rede e é responsável pelos objectos de aplicação. Os objectos de aplicação permitem personalizar um dispositivo, permitindo desta forma o controlo dos protocolos das outras camadas. Esta camada é também responsável pelo endereçamento da rede, criando perfis e *clusters* [16].

II.6.8.Roteamento

Nas redes *ZigBee*, utilizando transmissões *unicast*, será necessário utilizar roteamento quando um pacote necessitar de passar por mais de um *router* até chegar ao seu destino. Existem três tipos de roteamento, dependendo das necessidades e características de cada rede [12]:

O roteamento em malha utiliza uma tabela de roteamento em cada dispositivo. Desta forma, são criados caminhos entre a origem e o destino, possivelmente através de vários saltos. Todos os dispositivos sabem para quem mandar os pacotes até ao destino final. Isto é útil em redes que não seja necessário efectuar roteamento em muitos diferentes destinos, permitindo que todos os destinos podem ser inseridos na tabela de roteamento.

O roteamento muitos-para-um significa que uma única transmissão *broadcast* é efectuada, configurando rotas em todos os dispositivos para aquele que enviou o *broadcast*. Este tipo de roteamento é útil quando temos muitos dispositivos remotos a enviar dados para o coordenador.

No roteamento por fonte, é incluído nos pacotes de dados todo o caminho a ser percorrido pelo pacote, para que este consiga atravessar a rede até chegar ao destino. Isto é utilizado de forma a melhorar a eficiência em grandes redes.

II.7. MODELO DE COLABORAÇÃO NAS REDES DE SENSORES SEM FIOS

O WSNSCW - “Wireless Sensor Networks Supported Cooperative Work” – é um modelo colaborativo criado especificamente para redes de sensores sem fios, considerando os requisitos particulares deste tipo de rede. Através deste modelo é possível mostrar a colaboração existente na rede e também modelar toda a rede de sensores, assim como todas as suas entidades (os diferentes nós, o estado dos nós, a bateria dos nós, a existência de clusters, as estações base, sessões, tipos de dados recolhidos, tipos de sinais transmitidos), as suas propriedades, relações, estados, entre outros.

Este modelo também permite uma visualização da rede de uma forma hierárquica, assim como a representação de cada estado da rede e a sua evolução [3].

II.7.1. Definições das entidades

As entidades são todos os diferentes componentes que compõem a rede de sensores sem fios.

Neste tipo de rede temos dois tipos de nós diferentes: os *sensor nodes* (que podem estar em dois estados, activo ou inactivo - modo *sleep*), os *anchor nodes* (que suportam os *sensor nodes* no processo de localização), os *sink nodes* (também conhecidos como *base stations*, que são responsáveis pelo envio de dados até o *gateway*) e o *gateway* (que é responsável pelo envio de dados para o utilizador, através de internet ou satélite).

Um *cluster* é um grupo de nós, que pode ter sido criado de diversas formas: de acordo com a sua área geográfica, tipo de *sensor nodes*, tipo de fenómeno, tarefa a ser executada, entre outras. Se os nós estão agrupados em *clusters*, um dos seus membros será o *cluster head* (sendo que existe apenas um *cluster head* por *cluster*). Desta forma, todos os nós do cluster depois de recolherem os dados, transmitem essa informação para o *cluster head*, que por sua vez a envia para o *sink node*.

Uma *relationship* é criada quando dois nós colaboram. Nesta colaboração são partilhados dados tais como a temperatura, humidade, luz, gases poluentes, que podem ser enviados para outros nós utilizando outros tipos de sinais (rádio, ultrassom, entre outros). Os *obstacles* são objectos (edifícios, árvores, rochas) que podem obstruir a linha de sinal entre os nós, não permitindo desta forma uma comunicação directa entre ambos, afectando desta forma a forma como as *relationships* são criadas.

Podem ser criadas *collaborative sessions* quando se efectua a monitorização de uma rede de sensores sem fios, que podem serem executadas em simultâneo. Estas sessões podem ser estabelecidas baseando-se em novos objectivos.

A *battery* é um dos recursos mais importantes do *sensor node*, assim, é muito importante o *user* saber qual a bateria de cada nó. É por isto que a *battery* também é uma entidade do modelo.

Finalmente, o *user*, que é a entidade que interage com a rede de sensores sem fios, definindo a aplicação, visualizando dados, personalizando o trabalho dos *sensor nodes*, entre outros [3].

II.7.2. Hierarquia de colaboração nas redes de sensores sem fios

Nas redes de sensores sem fio existem dois tipos de participantes, os humanos e os nós. Os vários tipos de nós colaboram entre si de forma a completarem as suas tarefas. A estação base colabora com os nós para receber os seus dados. Para os utilizadores comunicarem com a rede, utilizam a estação base, que por sua vez comunica com os vários nós da rede de forma a efectuar a tarefa que o utilizador definiu [13].

II.7.3. Tipos de Colaboração

Os tipos de colaboração numa WSN dependem do tipo de dispositivo envolvido nessa colaboração. Geralmente, a colaboração ocorre entre nós que estão localizados numa certa região, o que significa que o grupo de nós que colabora poderá não ser sempre o mesmo, visto que se um nó sair dessa região, deixará de colaborar com os outros nós. Os nós poderão também colaborar através da monitorização de um aspecto comum a todos eles [13].

II.7.3.1. Colaboração entre o utilizador e a rede de sensores sem fios

O utilizador é a entidade que interage com a rede, definindo a aplicação, efectuando pesquisas na rede, visualizando os dados, interagindo com os nós da rede, etc. Os nós também poderão iniciar a comunicação em alguns casos, visto que algumas mudanças no ambiente poderão levar a que os nós mandem alertas para os utilizadores. Desta forma podemos verificar que este tipo de colaboração ocorre nos dois sentidos.

O utilizador comunica com o nó através de um computador ou de um PDA, ou seja, através de algum dispositivo que possua o protocolo utilizado na rede. Os

nós comunicam com o utilizador através da estação base, que recebe os dados dos outros nós sensores da rede.

II.7.3.2.Colaboração entre os nós sensores

Existem diversas maneiras de existir colaboração entre os nós sensores. Poderá existir colaboração entre sensores do mesmo tipo (entre nós sensores ou entre nós âncora por exemplo), existe também colaboração na hierarquia normal da rede, ou seja, entre os nós sensores e os nós âncora, e entre os nós sensores e a estação base. Esta colaboração visa a partilha de dados.

Se aplicarmos algum algoritmo de *clustering*, ficamos com os vários nós divididos em clusters, ou seja, ficam agrupados consoante a sua função por exemplo. Ao formarmos clusters, é necessário definir um cluster *head*, ou seja, definir um nó a que os outros nós irão mandar os seus dados. Desta forma, temos colaboração entre o cluster *head* e os vários nós que pertencem a esse cluster, visto que é o cluster *head* que irá transmitir a informação para a estação base, ou para outro cluster *head*. Assim, temos colaboração entre os vários clusters *head*, e também entre o cluster *head* e os nós sensores desse cluster.

II.7.3.3.Colaboração entre nós do mesmo tipo

Existem diversos tipos de colaboração entre nós do mesmo tipo, sendo que podemos identificar várias:

- Colaboração ponto-a-ponto: Os colaboradores têm os mesmos papéis e as mesmas funções na colaboração. Este tipo de colaboração pode ocorrer entre nós vizinhos, entre um grupo de nós em estado activo ou entre um grupo de nós que está a monitorizar um fenómeno comum.
- Colaboração mestre-escravo: No processo de colaboração, o nó “mestre” coordena o trabalho de todos os nós “escravos” da rede e partilha alguma informação acerca da colaboração. Os nós “escravos” apenas têm a tarefa de executar o que foi pedido pelo mestre. No caso da utilização de *clustering*, a colaboração é estabelecida entre o *cluster head* e os outros nós sensores que pertencem ao cluster.
- Colaboração produtor-consumidor: Neste tipo de colaboração, o nó produz informação que irá ser consumida por outros nós, independentemente do nível desses nós na hierarquia de colaboração. Este tipo de colaboração pode acontecer tanto na colaboração ponto-a-ponto ou mestre-escravo.

II.7.3.4.Colaboração entre *sensor nodes* e *anchor nodes*

Neste tipo de rede podem existir vários tipos de colaboração de forma a atingir o objectivo pretendido. Neste caso, e de forma a determinar a localização dos *sensor nodes*, também existe colaboração entre os *sensor nodes* e os *anchor nodes* de forma a que os próprios sensores saibam a sua localização, sem intervenção humana.

A utilização de um algoritmo de localização envolve a utilização de *anchor nodes*. A utilização de um algoritmo destes é só utilizado nas redes em que o *deployment* é *ad hoc*. Nas redes em que a posição dos nós é colocada manualmente, a localização dos nós é conhecida.

II.7.3.5.Colaboração entre *sensor nodes* e o *sink node*

Neste tipo, existe colaboração em ambos os sentidos, visto que todos os *sensor nodes* recolhem e enviam os dados para o *sink node* e também o *sink node* comunica com os *sensor nodes* quando envia comandos para os mesmos. Caso seja aplicado um algoritmo de cluster, este tipo de comunica já não existe, passando a existir colaboração entre os *cluster heads* e o *sink node*.

II.7.3.6.Colaboração entre *sensor nodes* e *cluster head*

Ao aplicar o algoritmo de *clustering*, um dos membros do cluster torna-se *cluster head*. Este tipo de colaboração envolve:

- O envio de dados dos *sensor nodes* do cluster para o *cluster head*;
- O *cluster head* recebe comandos do utilizador e envia-os para os *sensor nodes*;
- Os *sensor nodes* e o *cluster head* têm que ir trocar informação frequentemente de modo a que o *cluster head* possa actualizar quais os elementos que fazem parte do seu cluster (alguns podem ter saído do *cluster* devido a falta de bateria ou terem sido movidos para outros *clusters*).

II.7.3.7.Colaboração entre *cluster head* e *sink node*

Se for aplicado um algoritmo de *clustering*, o *cluster head* é o responsável pela recolha de dados de todos os *sensor nodes* do *cluster* e pelo envio desses dados para o *sink node*. Nesta colaboração, o *sink node* também pode enviar comandos para o *cluster head*. Esta conexão entre os dois tem que ser garantida, e quando o *cluster head* não tem largura de transmissão suficiente, o esquema de *clustering* estabelece uma

relação *inter-cluster heads* de forma a encaminhar os dados de cada *cluster head* para o *sink node*.

II.7.3.8. Colaboração entre sessões

A colaboração entre sessões acontece quando alguma informação tem que ser passada entre sessões, como por exemplo uma sessão começar apenas quando outra termina. Este tipo de colaboração também pode ocorrer quando uma sessão tem o seu início baseado num fenómeno que é detectado noutra sessão. Ao observar a Figura II.6, podemos verificar como acontece a colaboração na WSN. Num nível mais baixo, existe colaboração entre os NOS, que enviam os ATE ao *cluster head* (no caso de ter sido aplicado um algoritmo de *clustering*), que por sua vez estão englobados numa sessão. O utilizador depois, tem acesso a todos os dados que são recolhidos da rede, sendo o nível mais alto da colaboração numa rede de sensores sem fios.

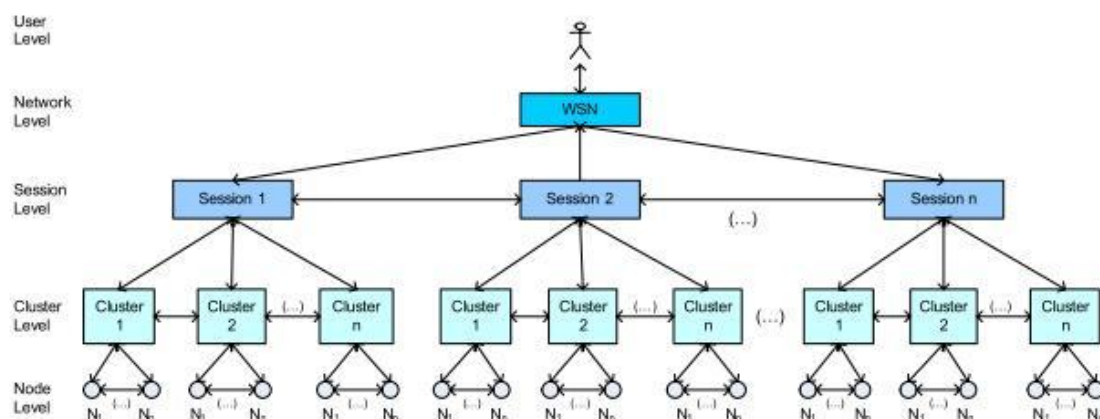


Figura II.6: Diferentes níveis hierárquicos da colaboração na WSN [13].

II.7.4. Papéis dos utilizadores

Temos vários participantes nas redes de sensores sem fios, o utilizador, o nó sensor, o nó âncora, a estação base e o *cluster head*, que é utilizado quando existe a aplicação de um algoritmo de *clustering*.

II.8. SESSÕES COLABORATIVAS

Uma sessão é a unidade essencial num trabalho colaborativo nas redes de sensores sem fio. É criada com base nos objectivos do utilizador, que podem ser o tipo de fenómeno a observar, a área a efectuar a monitorização (escolhendo os nós dessa área), o tempo de monitorização, entre outros.

As sessões colaborativas são compostas por vários participantes, o utilizador, o nó sensor, o nó âncora, a estação base e o cluster *head*, que é utilizado quando existe a aplicação de um algoritmo de *clustering*. Estes tipos de nós colaboram entre si de forma a realizar as tarefas.

Uma sessão pode ter vários estados consoante o estado dos nós que estão nessa sessão:

- *Created*: A sessão encontra-se neste estado quando foi criada mas não foi iniciada, ou seja, ainda não se encontra no estado open. Este é o primeiro estado de todas as sessões.
- *Open*: A sessão está neste estado se o objectivo pelo qual foi criada ainda não estiver chegado ao fim, ou seja, os nós que compõem esta sessão ainda se encontram activos. Se uma sessão encontra-se activa, significa que existe comunicação entre dois dispositivos, desta forma, podemos concluir que quando a sessão está activa, existe comunicação.
- *Close*: Uma sessão está no estado inactivo se todos os nós que compõem a sessão estão no estado *sleep*, quando existe algum problema com todos os nós que estão na sessão, por exemplo, estão todos sem bateria ou quando a sessão pára de monitorizar num tempo que foi especificado pelo utilizador.
- *End*: A sessão está neste estado quando acaba o objectivo para a qual foi criada.

As sessões também podem ser classificadas de acordo com a ordem temporal das mesmas. Podem ocorrer em paralelo (quando ocorrem ao mesmo tempo) ou em sequência (uma sessão começa quando a outra acaba), de forma síncrona (quando a sessão é planeada pelo utilizador) ou de forma assíncrona (a sessão é iniciada devido a alguma mudança inesperada no ambiente, ou seja, não é o utilizador a definir quando a sessão começa).

II.9. TRABALHOS RELACIONADOS

Depois de uma revisão bibliográfica, não foram encontradas ferramentas que permitissem realizar o mesmo trabalho que a ferramenta que foi desenvolvida neste projecto, ou seja, que permitissem criar, monitorizar e gerir sessões colaborativas em WSN. Desta forma, foi feita uma comparação com plataformas que efectuassem a recolha e visualização das amostras recolhidas das redes de sensores sem fios, e com ferramentas que permitissem efectuar a gestão da rede, para poder posicionar a plataforma proposta. Em seguida, serão apresentadas as ferramentas mais relevantes encontradas na literatura.

II.9.1.Mote View

Esta ferramenta permite efectuar a visualização e gestão de uma rede de sensores sem fios. O Mote-View permite visualizar os dados das amostras instantaneamente através de uma folha de dados ou através de gráficos. Esta ferramenta permite efectuar a visualização de todos os nós da rede, e verificar o seu *status* e endereço de 64-bit.

Com esta ferramenta podemos visualizar as últimas leituras efectuadas pelos sensores, visualizando dados como o *id* do nó, a hora que a amostra foi recebida e também os valores que foram retirados dos sensores.

Utilizando o Mote-View, podemos visualizar a topologia da rede, conseguindo desta forma verificar a localização dos nós [14].

II.9.2.MonSense

É um *software* que ainda está a ser desenvolvido, que irá permitir fazer o *deployment*, monitorização e controlo de várias redes de sensores sem fios. De seguida, mostra-se as funções deste *software* em termos de visualização e controlo da rede.

Em termos de visualização, através deste *software* será possível visualizar, em tempo real, os dados recolhidos pelo sensor através de um mapa colorido. Também será possível visualizar os dados das amostras recolhidas anteriormente, num ficheiro MonSense log.

No controlo da rede, podemos definir a latitude, longitude e altitude de cada nó. Também será possível enviar comandos para a rede, para efectuar *reset* à rede ou iniciá-la [15].

II.9.3.Tiny DB

Com a utilização desta *software*, é possível efectuar interrogações à rede, para saber quais os id's dos nós da rede, e quais os valores dos parâmetros que a rede monitoriza. Estas interrogações são feitas através de consultas, utilizando uma linguagem semelhante ao *SQL*. Permite também a execução de alguns comandos, que possibilitam ao utilizador efectuar um *reset* à rede, fazer e desfazer uma comunicação e também visualizar a topologia da rede [18].

II.9.4.MANNA – Management Architecture for *Wireless Sensor Networks*

A MANNA é uma arquitectura de gestão de redes sem fios. Permite a visualização da topologia da rede através da verificação da conectividade dos nós. Além disto, mostra um mapa com o nível de bateria dos nós da rede e fornece um mapa com a região que os sensores estão a monitorizar. Permite também enviar comandos para a rede, assim como escolher uma área de monitorização [14, 30].

II.9.5.BOSS – Bridge Of SensorS

BOSS é uma ferramenta que efectua a gestão da rede, baseando-se no envio de comandos para descobrir a mesma. Esta ferramenta fornece uma ponte entre a rede de sensores sem fios e o utilizador, permitindo saber informações do estado dos nós sensores, efectuar a gestão do nível de energia da rede, saber a localização dos nós sensores e permitir receber amostras da rede em tempo real [14, 31].

II.9.6.Comparação das diversas ferramentas

Para efectuar uma análise destas ferramentas foram estabelecidos alguns parâmetros de comparação. Na Tabela II.3, iremos comparar as ferramentas encontradas em termos de capacidade de monitorização em tempo real, visualização da rede, a capacidade de adicionar nós automaticamente e visualizar as relações entre eles.

Tabela II.3: Comparação em relação aos parâmetros de visualização.

Visualização Ferramenta	Monitorização em tempo real	Visualização dos nós da rede	Visualização da topologia da rede	Descoberta automática de nós
Tyni DB	Sim	Sim	Sim	Sim
MonSense	Sim	Sim	Sim	Sim
Mote-View	Sim	Sim	Sim	Não
MANNA	Sim	Sim	Sim	Sim
BOSS	Sim	Sim	Sim	Sim

Visualizando a tabela, verificamos que as ferramentas são similares, e que todas elas cumprem os requisitos mínimos para uma ferramenta de visualização de redes de sensores sem fios.

Na parte de gestão da rede, iremos explorar se estas ferramentas têm capacidade de utilizar sessões colaborativas, criar um histórico das amostras recebidas, efectuar a gestão da rede (através de comandos) e de gerar relatórios dos dados recebidos. A comparação destes parâmetros está descrita na Tabela II.4.

Tabela II.4: Comparação em relação aos parâmetros de gestão de rede.

Gestão Ferramenta	Histórico	Gestão da rede	Gerar Relatórios	Criar Sessões	Consultar Sessões	Monitorizar sessões
Tiny DB	Sim	Não	Não	Não	Não	Não
MonSense	Sim	Sim	Não	Não	Não	Não
Mote-View	Sim	Sim	Sim	Não	Não	Não
MANNA	Não	Sim	Não	Não	Não	Não
BOSS	Não	Sim	Não	Não	Não	Não

Ao comparar as diversas ferramentas, verificamos que a maior parte permite ao utilizador criar um histórico de amostras. Na gestão de rede, que engloba fornecer uma interface de comandos para comunicação com a rede, apenas a ferramenta Tiny DB não possui, sendo que concerne à geração de relatórios, apenas a ferramenta Mote-View os permite. No que toca às sessões colaborativas, nenhuma delas permite criar ou consultar as mesmas.

Existem situações em que é interessante efectuar a monitorização da rede de forma diferente, utilizando somente os nós de uma determinada área, monitorizando apenas um fenómeno em particular, ou monitorizar apenas durante um período de tempo. A Tabela II.5 mostra como as ferramentas analisadas dão respostas a estes itens.

Tabela II.5: Escolher o que monitorizar na WSN.

Monitorização Ferramenta	Escolher os nós sensores	Escolher fenómeno	Escolher o intervalo de tempo	Escolher a área a monitorizar
Tiny DB	Não	Não	Não	Não
MonSense	Não	Não	Não	Não
Mote-View	Não	Não	Não	Não
MANNA	Não	Sim	Não	Sim
BOSS	Não	Não	Não	Não

Ao comparar as diversas ferramentas, verificamos que a maior parte não permite ao utilizador definir a monitorização, ou seja, apenas recebem as amostras da rede. Desta forma, verificamos que efectuem a monitorização sem qualquer contacto com o utilizador.

Ao interrogar a rede de sensores sem fios, podemos ter acesso a todas as informações da mesma, desde as mais básicas, até todos os detalhes sobre os nós e as relações entre eles. Na Tabela II.6 encontra-se descrito o que conseguimos interrogar da rede, ao utilizar estas ferramentas.

Tabela II.6: Interrogar a rede.

Interrogar Ferramentas	Informações da WSN	Dispositivos da WSN	Topologia da Rede	Gestão da Energia
Tiny DB	Não	Sim	Não	Não
MonSense	Não	Não	Não	Não
Mote-View	Não	Não	Não	Não
MANNA	Não	Sim	Sim	Sim
BOSS	Sim	Sim	Sim	Sim

Depois de analisar estas ferramentas, verificamos que não existe uma ferramenta completa, ou seja, que consiga preencher todos os requisitos de visualização e gestão da rede, utilização de sessões colaborativas para a monitorização da mesma, ou seja, não oferecem ao utilizador a possibilidade de colaborar com a rede no sentido de escolher como, onde e quando utilizada a monitorização.

Na ferramenta proposta neste projecto de mestrado, irão ser incorporados alguns destes parâmetros de forma a construir uma ferramenta mais robusta e que permita dar resposta aos parâmetros mais importantes descritos anteriormente. Assim sendo, esta ferramenta irá focar-se mais na criação e gestão de sessões colaborativas nas redes de sensores sem fios.

II.10. CONCLUSÃO

As WSN são muito utilizadas em várias aplicações, como por exemplo na medicina e em dispositivos de automatização doméstica, devido ao seu baixo custo e facilidade de implementação da rede.

Ao utilizar o protocolo *ZigBee* garantimos que possuímos uma rede com baixas taxas de transmissão de dados mas de baixo custo, sendo ideal para ser utilizada em locais remotos, ou onde é necessário muitos dispositivos. Ao utilizar três tipos de dispositivos (coordenador, router e *end-device*), este protocolo permite ao utilizador criar diversas topologias, assim como aproveitar a utilidade dos *routers* no encaminhamento de dados. A topologia que permite usufruir de todas as capacidades dos dispositivos é a em árvore, em que temos um ponto central, o coordenador, temos os ramos, os routers, e finalmente, temos os *end-devices* que funcionam como folhas, efectuem a monitorização da rede.

Com a utilização do WSNCW - “*Wireless Sensor Networks Supported Cooperative Work*” - temos a representação de todas as entidades que são utilizadas na rede. Neste modelo é definido todos os tipos de colaboração existentes na rede, como por exemplo a colaboração entre sessões. Esta colaboração é efectuada quando alguma informação tem que ser transmitida entre as sessões. Estas sessões colaborativas podem estar em quatro estados diferentes: *Created* (quando a sessão é criada mas ainda não foi inicializada), *Open* (quando o objectivo da sessão ainda não foi concluído), *Close* (quando a sessão não se encontra a monitorizar) e *End* (quando todos os objectivos da sessão foram cumpridos, por exemplo, quando acaba o tempo especificado pelo utilizador para a monitorização).

Ao analisar as ferramentas relacionadas, foi verificado que a área de sessões colaborativas ainda está por explorar, visto que nenhuma ferramenta possibilitava a criação das mesmas.

No próximo capítulo irá ser explicado qual o contexto em que surge este projecto, assim como quais os requisitos e especificações necessários na ferramenta proposta.

III. DESENHO ARQUITECTURAL

III.1. INTRODUÇÃO

Neste capítulo será apresentado todo o desenho do sistema *WISE-MANager* e de que forma ele se enquadra no projecto *WISE-MUSE*, como mencionado no capítulo 1, visa a monitorização ambiental de museus, baseada em redes de sensores sem fios, para a conservação de obras de arte. Será igualmente descrito o sistema desenvolvido neste projecto de Mestrado, apresentando os requisitos funcionais e não funcionais, onde temos os requisitos de *hardware* e *software*, assim como os casos de utilização, onde podemos visualizar os actores e as suas tarefas no sistema. Será também apresentado um diagrama de sequência das sessões colaborativas, que irá mostrar como ocorre a mudança de estado. Finalmente, será igualmente apresentado o modelo de dados a utilizado e as suas entidades.

III.2. CONTEXTUALIZAÇÃO E DOMÍNIO DA APLICAÇÃO

Este projecto de mestrado, situado no contexto do projecto *WISE-MUSE*, tem como objectivo principal a criação e gestão de sessões colaborativas em redes de sensores sem fios. Com este projecto, a colaboração entre os sensores e o utilizador da rede será potencializado, permitindo ao utilizador efectuar o controlo e gestão da rede.

Este projecto foi desenvolvido para ser utilizado no Museu da Baleia, no entanto tem a capacidade de ser genérico para permitir a sua aplicação em uma *WSN* baseada no protocolo *ZigBee*.

A ferramenta proposta irá permitir a criação de sessões colaborativas, permitindo escolher qual o fenómeno a monitorizar, quais os nós que integrarão a sessão e quando ela iniciará. Dessa forma, o utilizador poderá definir o intervalo de monitorização. Em relação à visualização dos dados colectados, estes resultados podem ser apresentados em tabelas e históricos, que podem ser consultados. Além disto, o utilizador terá o controlo da rede, permitindo iniciar e parar a execução das sessões em qualquer momento.

III.3. REQUISITOS

Para que um projecto tenha sucesso, é muito importante saber, antes do seu início, quais os seus requisitos, tanto funcionais, como não funcionais. Para isto, é necessário estabelecer uma plataforma de entendimento entre todos os *stakeholders* (utilizadores do sistema) para obter uma definição correcta e explícita dos requisitos do sistema. Neste caso, os requisitos foram baseados no modelo *WSNSCW* [3] e nas necessidades específicas do Museu da Baleia.

III.3.1. Requisitos não funcionais

Os requisitos não funcionais são aqueles que contêm as restrições e qualidades do sistema. As qualidades são as propriedades ou características do sistema, sendo o que irá afectar o grau de satisfação dos utilizadores. As restrições, ao contrário das qualidades, não estão sujeitas a negociação e representam as necessidades do sistema para que este funcione de forma perfeita [19].

Existem diversos tipos de requisitos não funcionais, para este projecto iremos ter em conta requisitos de hardware e de *software*. De seguida são apresentados os requisitos que têm que ser cumpridos para a plataforma funcionar de forma correcta:

Requisitos de *Software*:

1. O sistema deve permitir a integração com a plataforma *WISE-MUSE*;
2. A implementação do projecto de criação de sessões colaborativas deverá ser realizada na linguagem de programação *Java* de forma a dar seguimento ao trabalho anterior;
3. O computador em que será utilizado o projecto terá que ter acesso à internet de modo a fazer a conexão com a base de dados da plataforma *WISE-MUSE*;
4. No caso de ser utilizado um servidor local, este sistema tem que ser executado no mesmo computador que recebe os dados das amostras dos fenómenos.

Requisitos de Hardware:

O sistema utilizará os módulos de radiofrequência *XBee*, assim como o protocolo *ZigBee*.

III.3.2. Requisitos Funcionais

Os requisitos funcionais são aqueles que suportam o sistema, descrevendo o que é suposto o sistema fazer, ou seja, as funções, tarefas ou comportamentos que o sistema tem que suportar. Existem diversos tipos de requisitos funcionais: usabilidade, fiabilidade, qualidade de serviço (performance), entre outros [20].

Para que este projecto tenha sucesso, é necessário que os seguintes requisitos sejam cumpridos:

1. O sistema tem que permitir criar, abrir, fechar e monitorizar sessões colaborativas;
2. Permitir criar uma sessão colaborativa com os seguintes dados: *ID* da sessão, descrição da sessão, horário de monitorização, nós da sessão e os parâmetros ambientais que a sessão monitoriza;
3. Permitir que o utilizador visualize as sessões criadas, e possa alterar os parâmetros;
4. Permitir ao utilizador manipular a sessão, ao permitir que comece a monitorizar imediatamente ou parar de monitorizar naquele momento;
5. Permitir que o utilizador verifique quais as sessões que estão a monitorizar em cada momento, assim como verificar quais as amostras dos parâmetros que cada sessão recolheu;
6. Permitir retirar informações úteis da rede, como por exemplo o *ID* da rede e o canal que a comunicação é efectuada;
7. Verificar quais os dispositivos que compõem a rede, e visualizar qual o nome e endereços de cada um dos dispositivos;
8. Permitir a alteração do nome dos dispositivos remotamente;
9. Permitir capturar as amostras enviadas pela rede;
10. Permitir a visualização dos dados capturados pela rede;
11. O sistema deve permitir que o utilizador elimine as sessões da base de dados;
12. Permitir ao utilizador exportar as amostras recebidas pela rede.

III.4. PROTÓTIPOS ABSTRACTOS CANÓNICOS

Os PACs são um modelo criado especialmente para suportar uma progressão suave desde a abstracção até ao desenho da interface para o utilizador. Desta forma, estes protótipos permitem mostrar a interface, e como os seus componentes encontram-se dispostos, assim como quais as opções que os utilizadores podem tomar [25].

Na Figura III.1 encontra-se o PAC que mostra como irão ser dispostos os componentes na criação de uma sessão colaborativa.

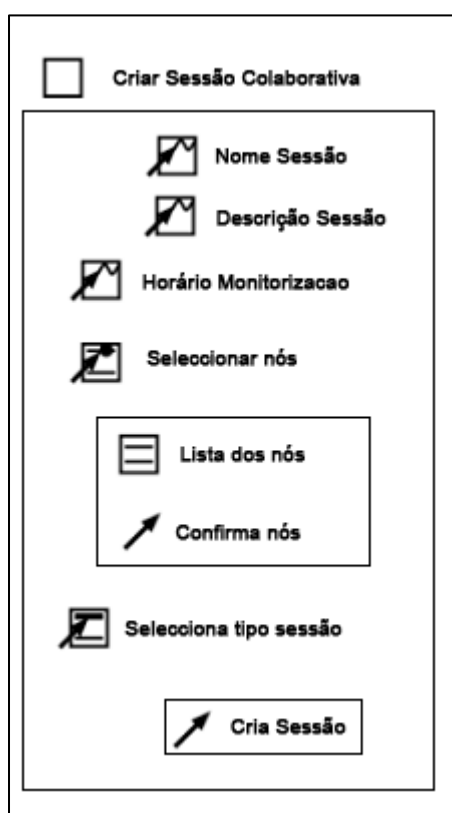


Figura III.1: PAC de criação de sessão colaborativa.

No PAC da Figura III.2, encontra-se a disposição dos componentes aquando da visualização das sessões activas, em que irá ser descrita cada sessão, o utilizador terá a possibilidade de visualizar os dados monitorizados, e quais os nós de cada sessão.

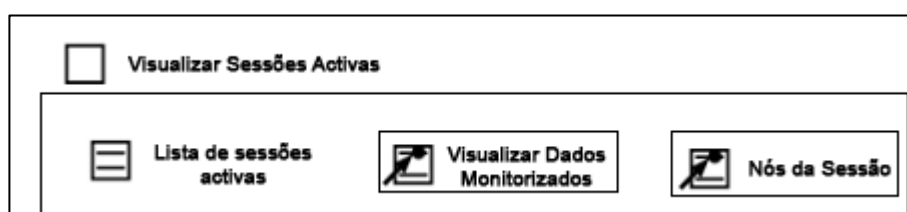


Figura III.2: PAC de visualização de sessões activas.

Na Figura III.3 encontra-se o PAC de resumo das sessões, onde podemos observar de que forma é mostrado ao utilizador os dados da sessão.

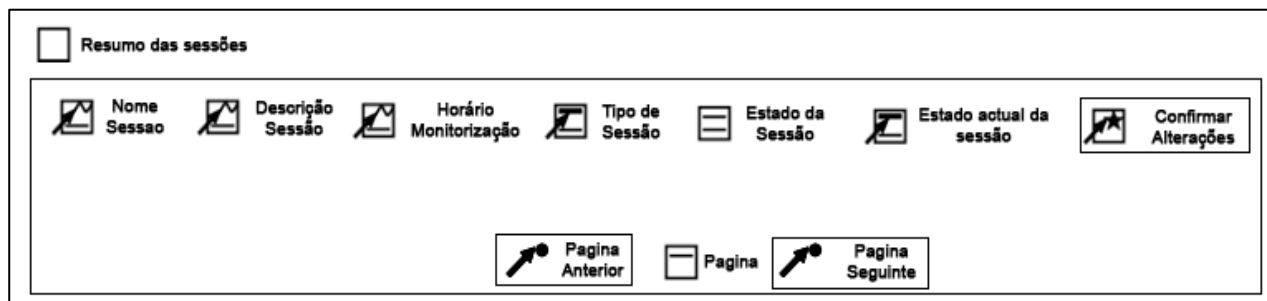


Figura III.3: PAC de resumo das sessões

Nas Figuras III.4 e III.5 estão os PACs de exportar dados da sessão e apagar sessões, respectivamente. Ao exportar dados, o utilizador escolhe de entre uma lista de sessões, qual a que quer ver o histórico, escolhendo também qual o intervalo de tempo das amostras. Para apagar sessões, apenas é necessário carregar no botão “Apaga Sessão”, na sessão desejada.



Figura III.4: PAC de exportar dados da sessão.

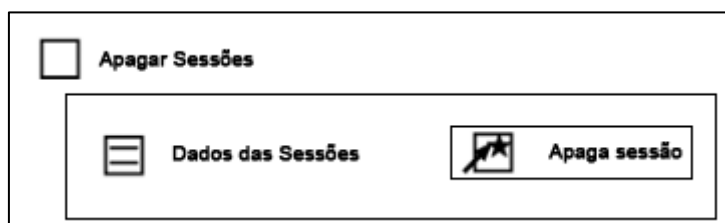


Figura III.5: PAC de apagar sessões.

Finalmente, nas Figuras III.6 e III.7, estão representados os PACs de informações da rede e nós da rede, respectivamente.

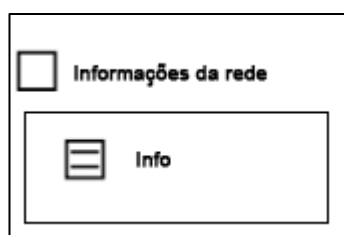


Figura III.6: PAC de informações da rede.

☐ Nós da rede

☒ Identificador Nó

☐ Endereço 16 bits

☐ Endereço 64 bits

☐ Endereço do Pai

☐ Tipo de dispositivo

☐ Número de filhos restantes

☒ Alterar identificador do nó

Figura III.7: PAC de nós da rede.

III.5. DIAGRAMAS UML

O UML, *Unified Modeling Language*, é uma linguagem standard utilizada para visualizar, especificar, construir e documentar todo o sistema. O UML será utilizado para documentar o sistema, mostrando os requisitos e o desenho do mesmo [24].

III.5.1. Diagrama UML Comportamental: Casos de utilização

Os casos de utilização descrevem a interacção, que pode ser provocada por um actor externo, entre o sistema e o ambiente. Um caso de utilização define um objectivo a ser cumprido, utilizando para isso um conjunto de interacções entre os actores externos e o sistema. Um caso de utilização é completado com sucesso quando o seu objectivo é satisfeito. Um actor é utilizado para descrever uma pessoa ou sistema que tem um objectivo a realizar. Os casos de utilização descrevem a forma como cada actor interage com o sistema [20].

Neste sistema temos os seguintes actores:

- O utilizador do programa, que cria as sessões colaborativas, e também efectua as outras funções do sistema.
- *WSN coordinator*, o coordenador da rede, que é um componente electrónico *XBee* que se encontra configurado para actuar como coordenador da rede de sensores sem fios, recebendo/enviando pacotes da/para a rede.
- Servidor e base de dados, onde irá ficar guardada toda a informação das sessões criadas e também das amostras recebidas pela rede.

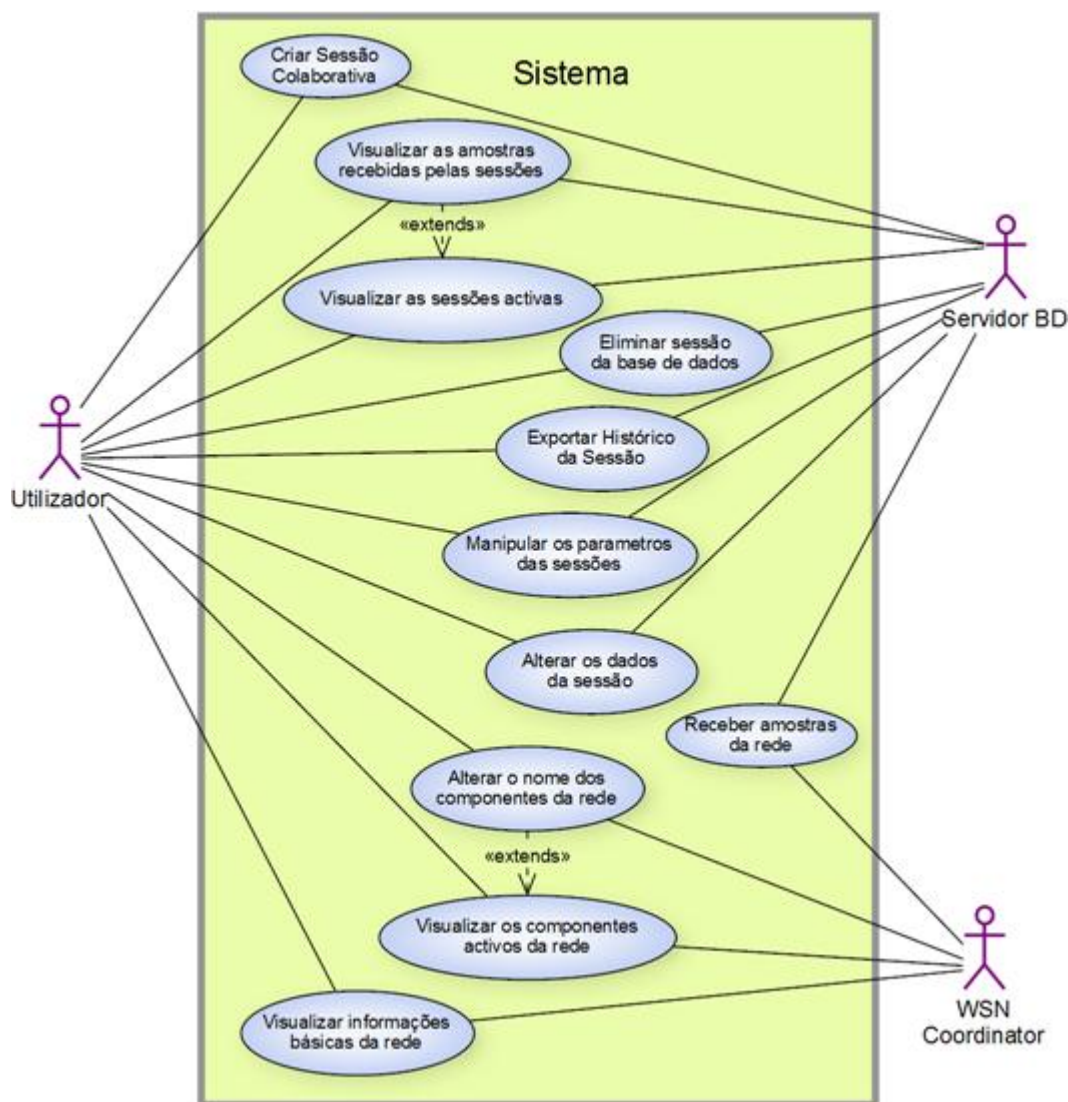


Figura III.8: Diagrama de casos de utilização.

Através da observação da Figura III.8, verificamos que temos dois casos de utilização que dependem de outros. O *Alterar o nome dos componentes da rede* só pode ser executado depois do caso de utilização *Visualizar os componentes activos da rede* e o *Visualizar as amostras recebidas pelas sessões*, que depende do *Visualizar as sessões activas*.

O *receber amostras da rede* permite receber as amostras enviadas pelo WSN Coordinator e são guardadas pelo Servidor BD.

Os casos de utilização apresentados na Figura III.8 irão ser detalhados na próxima secção através dos diagramas de actividades. Estes casos de utilização exemplificam as funcionalidades do sistema, desde a criação, visualização e manipulação das sessões colaborativas, até à gestão da rede por parte do utilizador.

III.5.2. Diagrama UML Comportamental: Diagrama de Actividades

Os diagramas de actividades são utilizados para mostrar de forma clara como é efectuada a circulação de dados no sistema. Desta forma é perceptível observar como o sistema irá reagir a determinadas situações, e visualizar de que forma irão ser implementadas as tarefas do sistema [21].

Assim, em seguida são apresentados os diagramas de actividade das tarefas referidas na Figura III.8.

III.5.2.1. Criar Sessão Colaborativa

A tarefa “Criar Sessão Colaborativa” é crucial na ferramenta, visto que é necessária para permitir que as sessões monitorizem os dados. De forma a efectuar esta tarefa, o utilizador inicializa a ferramenta e escolhe “*Create New Session*” no menu “*Session*”. Feito isto, aparece uma nova janela com todos os parâmetros que são necessários preencher para criar a sessão:

- *Name of session*: O nome da sessão;
- *Description of session*: Uma breve descrição da sessão;
- *Schedule of Monitoration*: Onde o utilizador preenche qual o horário em que a sessão estará no estado activo, ou seja, está a monitorizar;
- *Choose nodes for session*: Escolhe quais os nós que compõem a sessão colaborativa;
- *Session Type*: O tipo da sessão, ou seja, o que a sessão irá monitorizar.

Ao inserir os parâmetros, o sistema irá sempre dar feedback se os dados foram preenchidos incorrectamente. Não sendo permitido ao utilizador criar a sessão com os dados inválidos, aparecendo uma mensagem de erro. Na Figura III.9 está o diagrama de actividades desta tarefa.

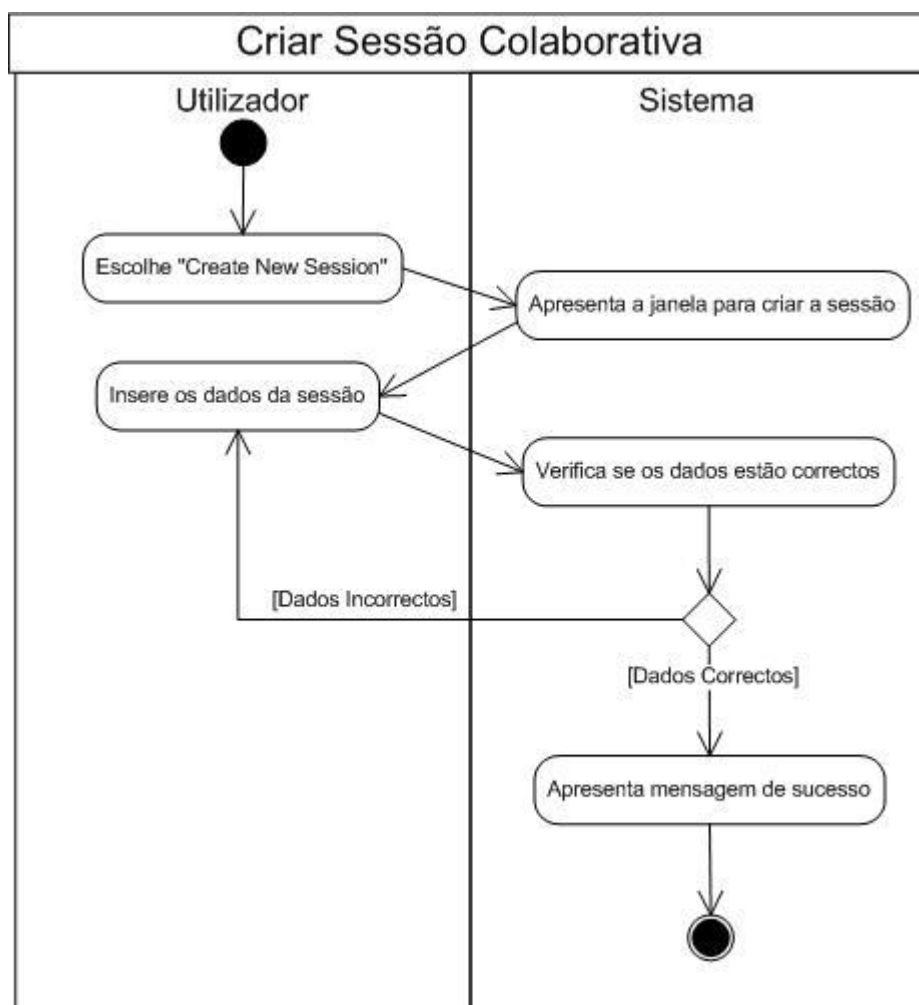


Figura III.9: Diagrama de actividade de "Criar Sessão Colaborativa".

III.5.2.2. Visualizar Sessões a Monitorizar

Para efectuar esta tarefa, é necessário clicar na opção "*View Active Sessions*", no menu "*Session*". É apresentada a lista das sessões que se encontram activas, ou seja, que estão a monitorizar a rede, mostrando informações como o nome da sessão, o horário de monitorização e os fenómenos a monitorizar.

O diagrama de actividade desta tarefa encontra-se na Figura III.10.

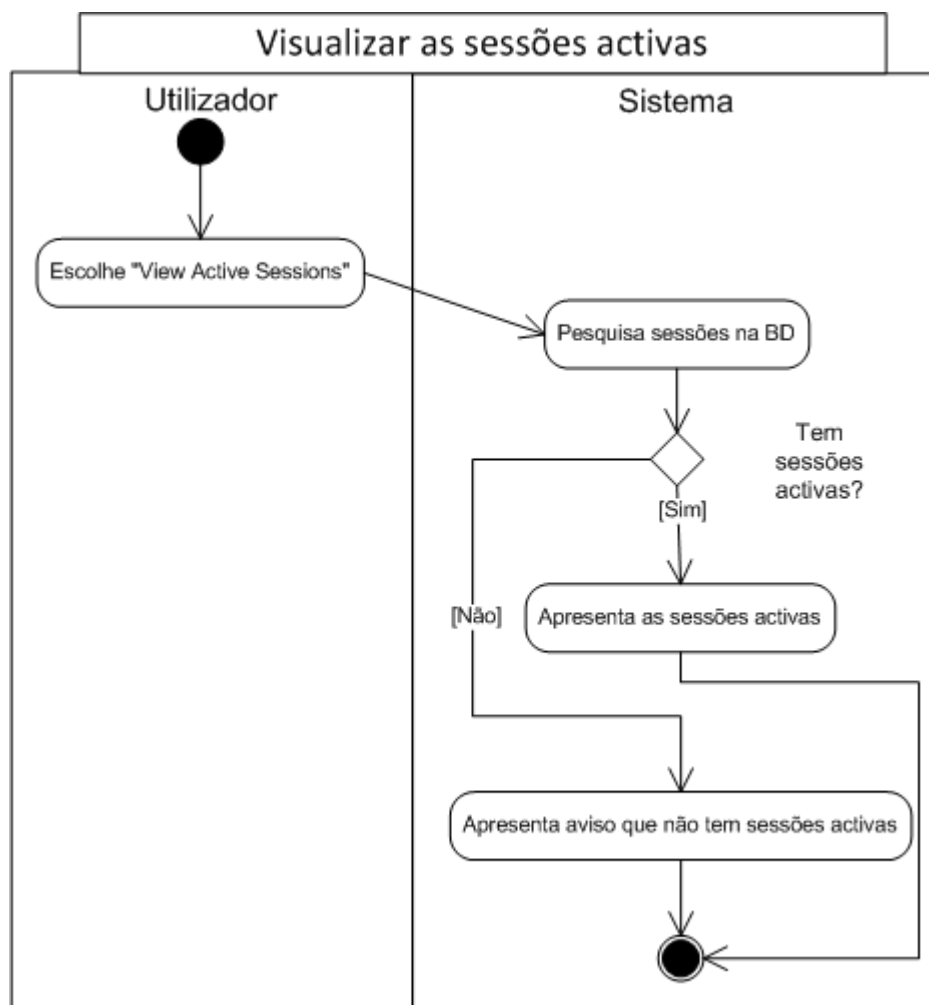


Figura III.10: Diagrama de actividade de "Visualizar sessões activas".

III.5.2.3. Visualizar Amostras de Dados

Através da Figura III.11 podemos observar como é que a tarefa *Visualizar Amostras de Dados* é executada. Para esta tarefa ser concluída, primeiro é necessário concluir a tarefa descrita na Figura III.10. Feito isto, para visualizar as amostras, é necessário clicar no botão "Monitoring Data", sendo que de seguida é apresentada uma nova janela com todas as amostras da sessão. Nesta janela é possível observar a data de monitorização, o nó que fez a monitorização, assim como as amostras dos fenómenos.

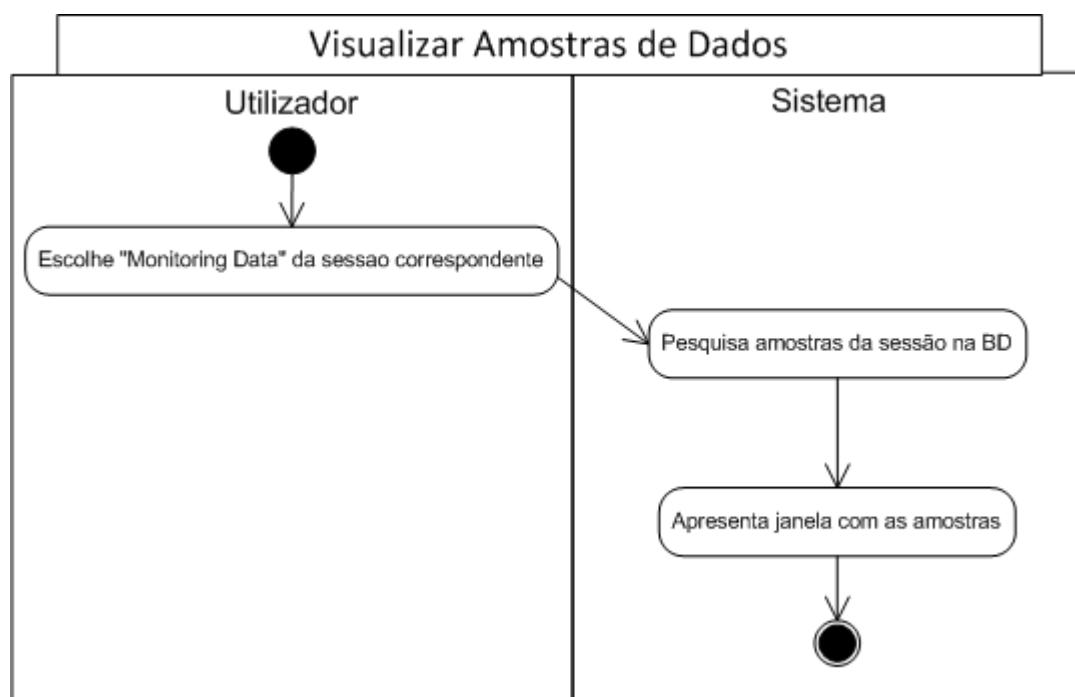


Figura III.11: Diagrama de actividade de “Visualizar amostras de dados”

III.5.2.4. Alterar Dados das Sessões

Nesta tarefa, descrita na Figura III.12, é necessário aceder à base de dados para recolher as informações de todas as sessões. Feito isto, o utilizador pode alterar os seus parâmetros, e quando terminar, guarda as alterações. Caso o utilizador altere parâmetros para valores incorrectos, será apresentada uma mensagem de erro.

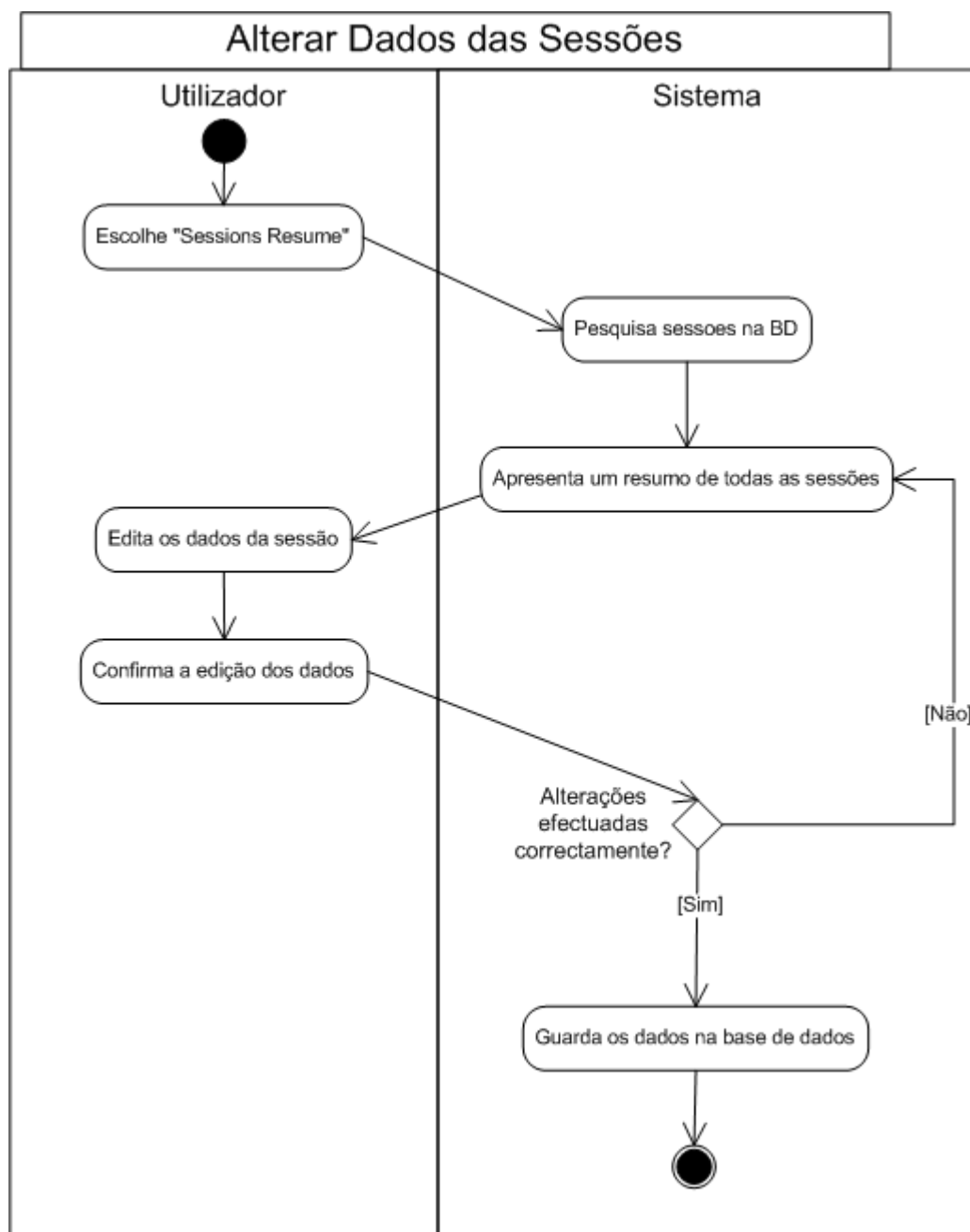


Figura III.12: Diagrama de actividade de "Alterar dados das sessões".

III.5.2.5. Visualizar Informações Básicas da Rede

O diagrama de actividades da tarefa "Visualizar informações básicas da rede" encontra-se na Figura III.13. Ao observar esta figura, verificamos que ao escolher a opção "WSN Info" no menu "WSN Management", o sistema irá verificar se a porta de comunicação com a WSN encontra-se aberta e só em seguida irá enviar os comandos para obter as informações básicas da rede, tais como, qual o PAN ID utilizado pela rede, qual o canal utilizado para efectuar a comunicação e as informações sobre os

dispositivos da rede, ou seja, quantos coordenadores, *routers* e *end-devices* existem na rede.

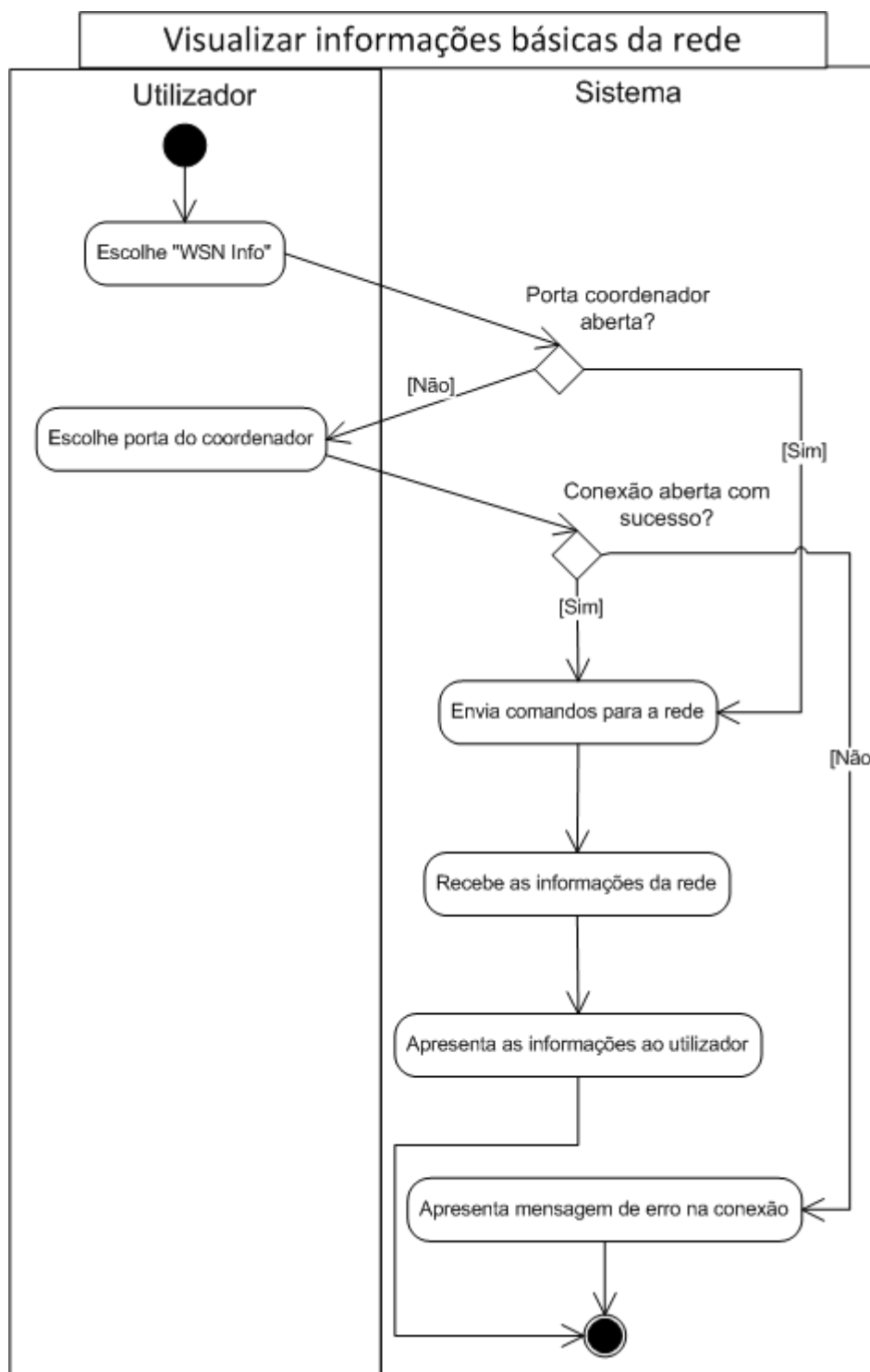


Figura III.13: Diagrama de actividade de “Visualizar informações básicas da rede”.

III.5.2.6. Visualizar os componentes activos da rede

Na Figura III.14 podemos observar o diagrama de actividade da tarefa “Visualizar os componentes activos da rede”. Para obter uma informação mais detalhada de todos os dispositivos da rede, é necessário escolher no menu “WSN Management” a opção “WSN Devices”. Desta forma, é possível ao utilizador observar dados como o nome dos dispositivos, os endereços de 64-bits e 16-bits, quantos nós ainda podem-se conectar a esse dispositivo para fazer parte da rede (apenas o coordenador e os routers têm esta opção), e finalmente, qual o tipo de dispositivo (coordenador, *router* ou *end-device*).

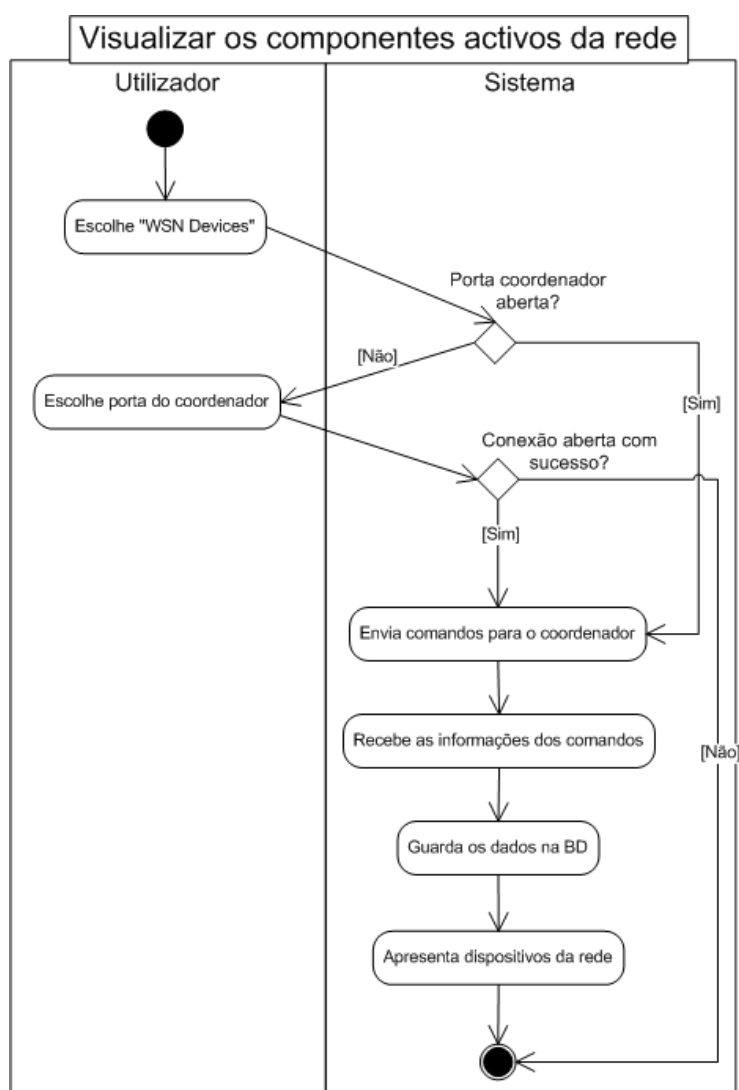


Figura III.14: Diagrama de actividade de “Visualizar componentes activos da rede”.

III.5.2.7. Alterar o Nome dos Dispositivos

Para alterar o nome do dispositivo, primeiro é necessário estar na lista dos dispositivos da rede, ou seja, é preciso efectuar a tarefa “*Visualziar componentes activos da rede*” primeiro. Depois, o utilizador alterar o nome do dispositivo desejado, e carregar no botão “*Change Node Ident*”. Ao carregar neste botão é enviado para o dispositivo em questão um comando com a alteração do identificador do nó, recebendo o utilizador uma confirmação desta alteração. O diagrama desta tarefa encontra-se na Figura III.15.

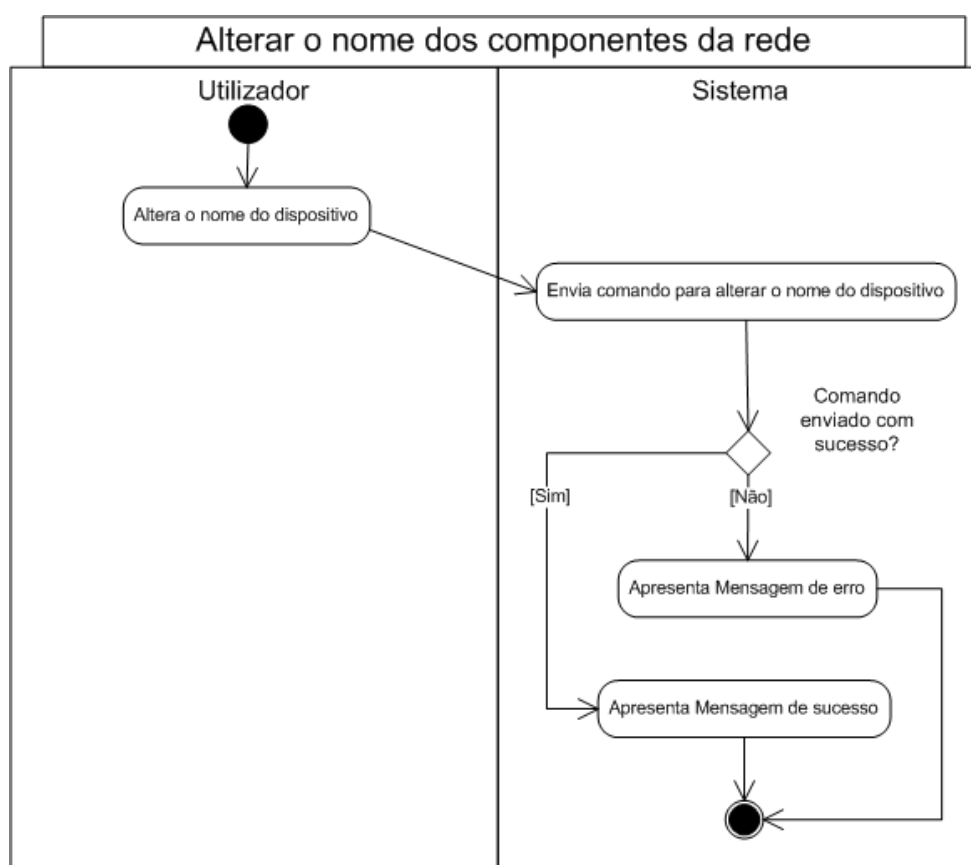


Figura III.15: Diagrama de actividade de “Alterar nome dos componentes da rede”.

III.5.2.8. Apagar Sessão

De forma a apagar sessões colaborativas, o utilizador terá que escolher no menu “*Session*” a opção “*Delete Sessions*”. Depois de o sistema efectuar uma pesquisa para verificar quais as sessões que estão guardadas, estas são apresentadas ao utilizador. O utilizador visualiza algumas informações das mesmas, tais como, o nome e descrição da sessão, o horário de monitorização e os fenómenos que cada sessão monitoriza. Ao lado destas opções encontra-se um botão “*Delete Session*” que ao ser activado irá proceder à eliminação da mesma. Para apagar a sessão, primeiro o

utilizador terá que confirmar que quer apagar todos os dados, e de seguida esta sessão é eliminada. Esta tarefa encontra-se descrita na Figura III.16.

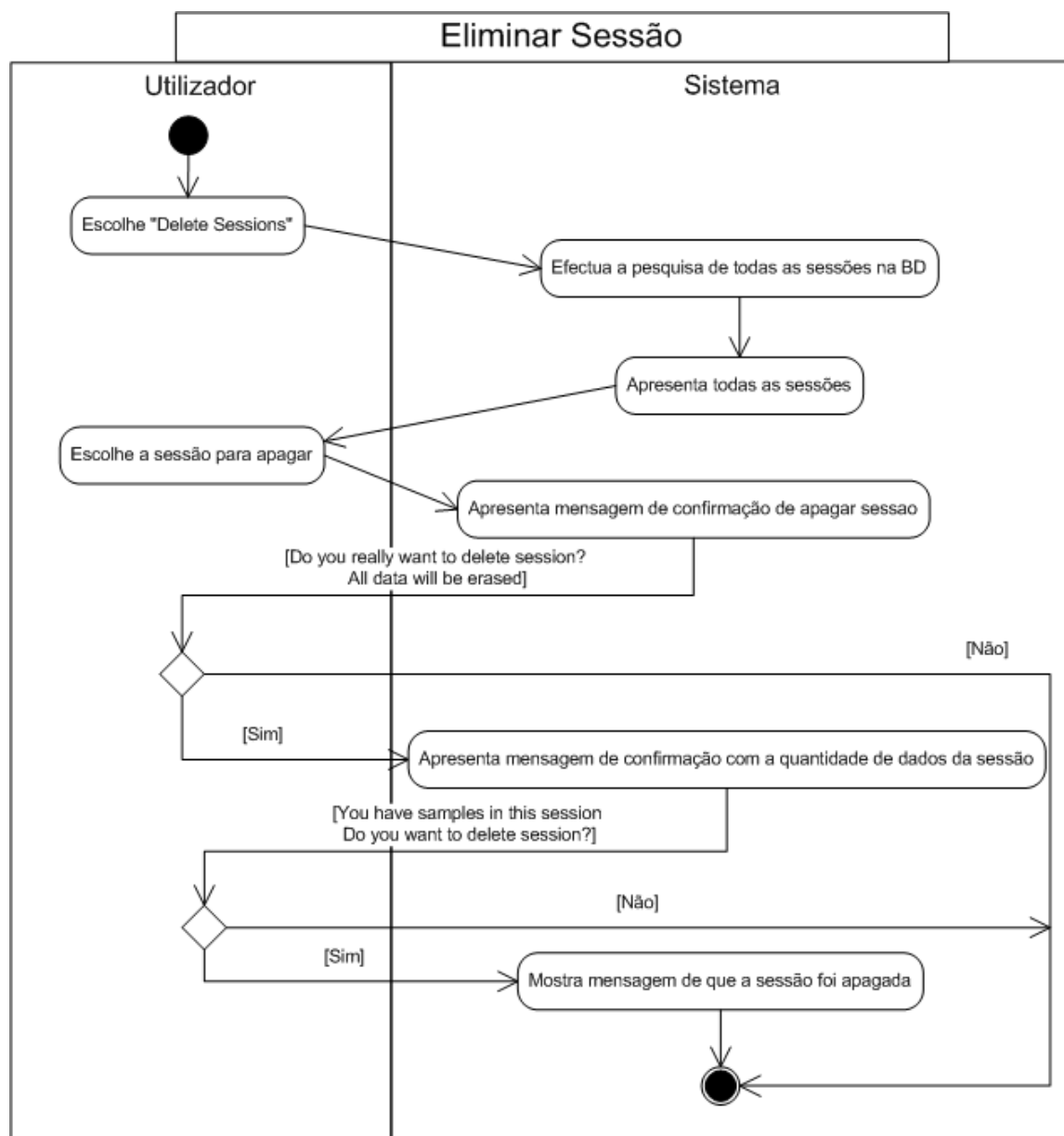


Figura III.16: Diagrama de actividade de “Apagar Sessão”.

III.5.2.9. Exportar Histórico de Sessões

A tarefa “Exportar Histórico de Sessões” tem o seu diagrama de actividade na Figura III.17. Neste diagrama podemos observar como decorre a circulação de dados quando o utilizador decide obter o histórico de alguma sessão.

Ao escolher a opção “*Export Data*”, que se encontra no menu “*Session*”, irá ser apresentado ao utilizador um formulário onde poderá escolher qual a sessão e qual o intervalo de tempo para gerar o relatório. Depois de escolher, apenas necessita de carregar em *Generate Session Historic*, sendo apresentada uma mensagem de erro ou de sucesso, sendo que neste caso, esta mensagem contém o local para onde foi gerado o relatório.

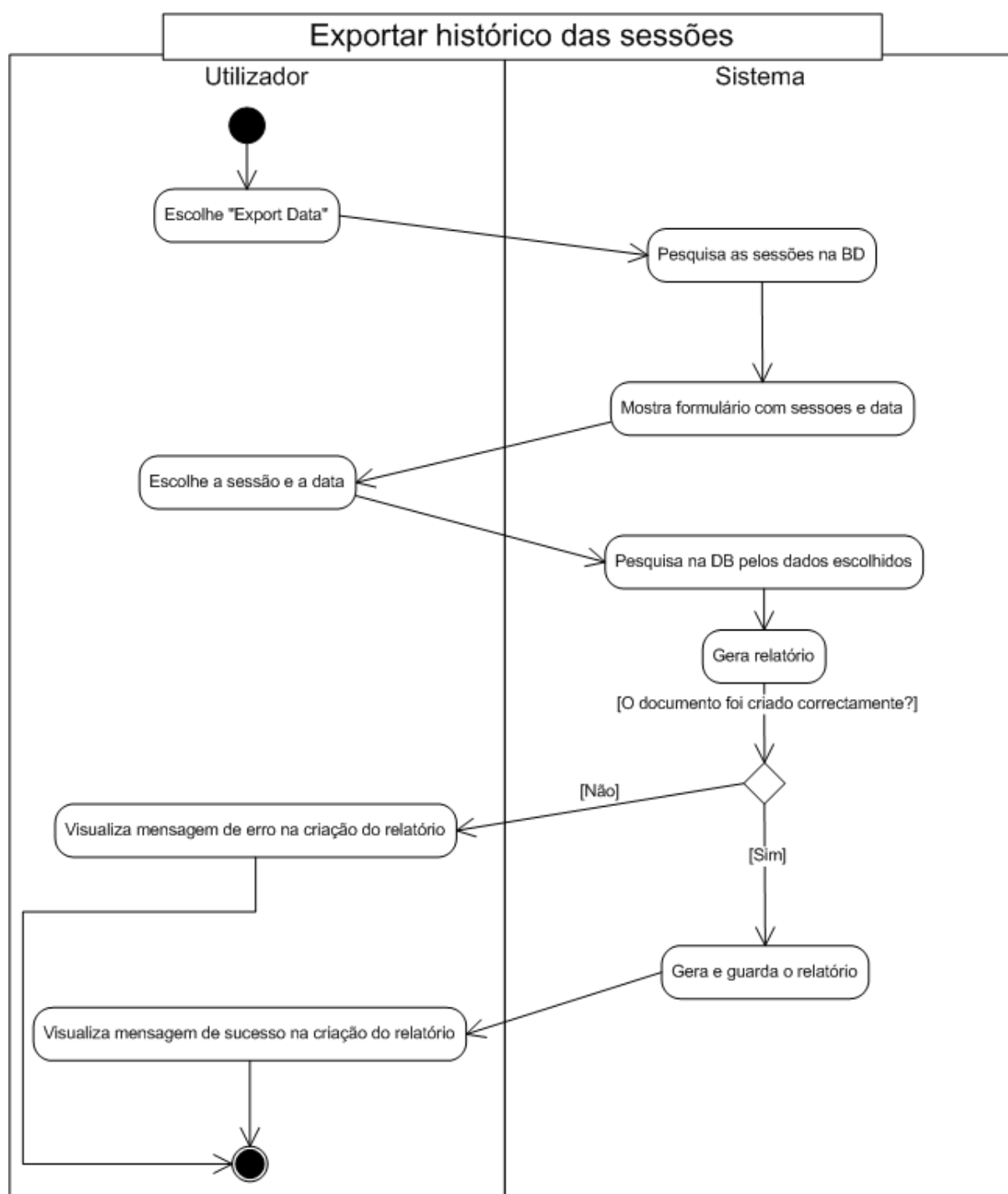


Figura III.17: Diagrama de actividade de “Exportar histórico das sessões”.

III.5.3.Diagrama UML de interacção: Diagrama de Sequência

Os diagramas de sequência, também chamados de diagramas de interacção, são utilizados para modelar os aspectos dinâmicos do sistema. Um diagrama de sequência mostra como é efectuada a interacção; consistindo entre um conjunto de objectos e as suas relações. Este tipo de diagrama é um diagrama de interacção que permite mostrar os aspectos dinâmicos de um sistema, mostrando como ocorre a colaboração entre classes, interfaces ou componentes apresentando as mensagens que são trocadas entre ambos [24].

III.5.3.1. Diagrama de Sequência das sessões colaborativas

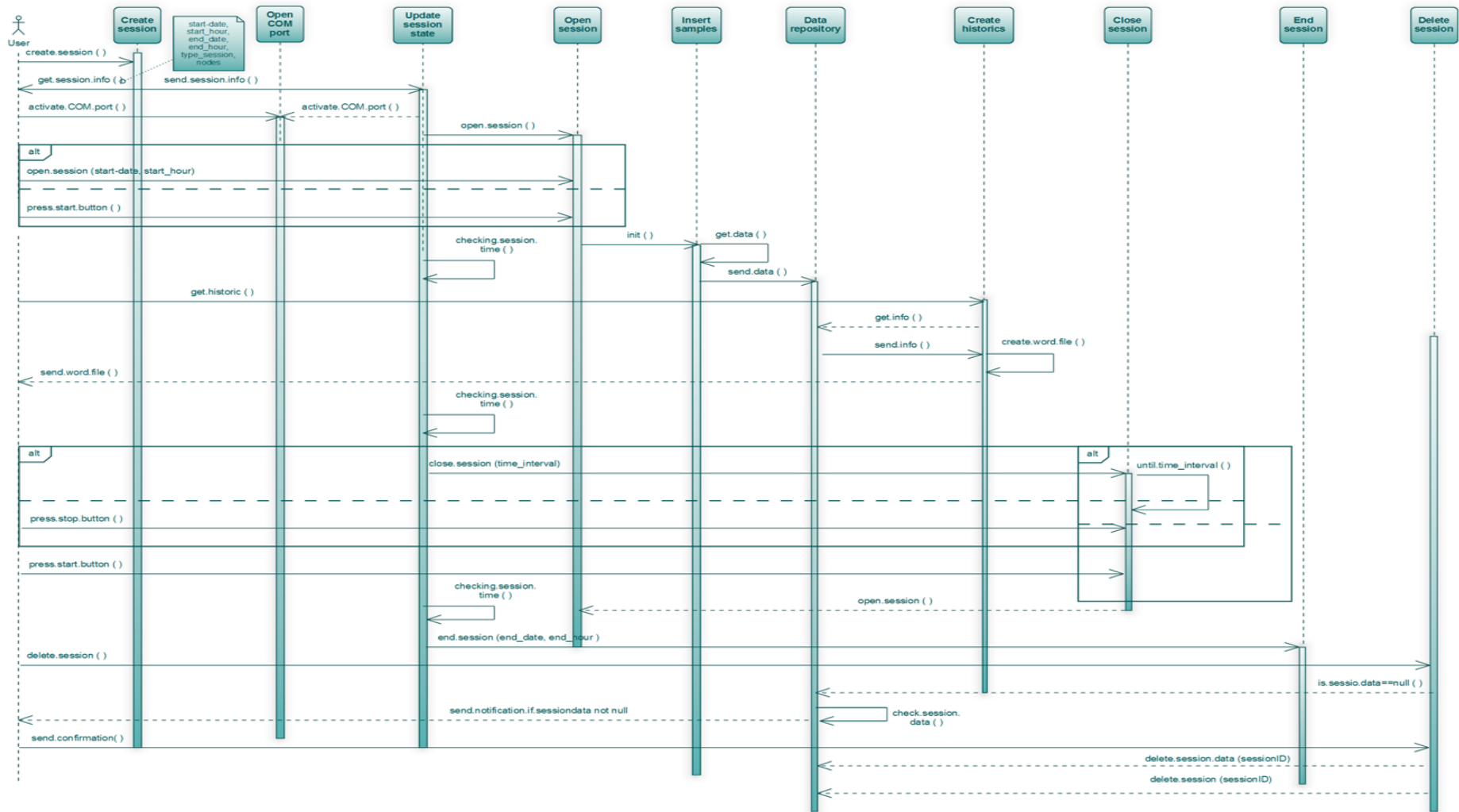


Figura III.18 Diagrama de Sequência das Sessões Colaborativas.

Ao observar a Figura III.18, verifica que esta representa quais as funções e módulos responsáveis pela criação e gestão de sessões colaborativas.

III.5.4. Diagrama UML Comportamental: Diagrama de transição de estados

Uma sessão colaborativa pode estar em quatro estados diferentes, *Created*, *Open*, *Close* e *End*. Ao criar a sessão, é necessário preencher diversos parâmetros, como podemos observar na Figura IV.18. Depois de criada, a sessão fica no estado *Created* até chegar ao seu horário de monitorização. Ao chegar a este horário, passa para o estado *Open* onde efectua a monitorização da rede. Depois de estar neste estado, só poderá passar para o estado *Closed* ou para o *End*. Passa para o estado *Close* quando a sessão não se encontra a monitorizar, ou seja, quando o utilizador decide parar a monitorização, ou quando acaba o tempo de monitorização. Finalmente, o estado *End* representa que a sessão acabou de monitorizar. Na Figura III.19 está representado o diagrama de estados de uma sessão.

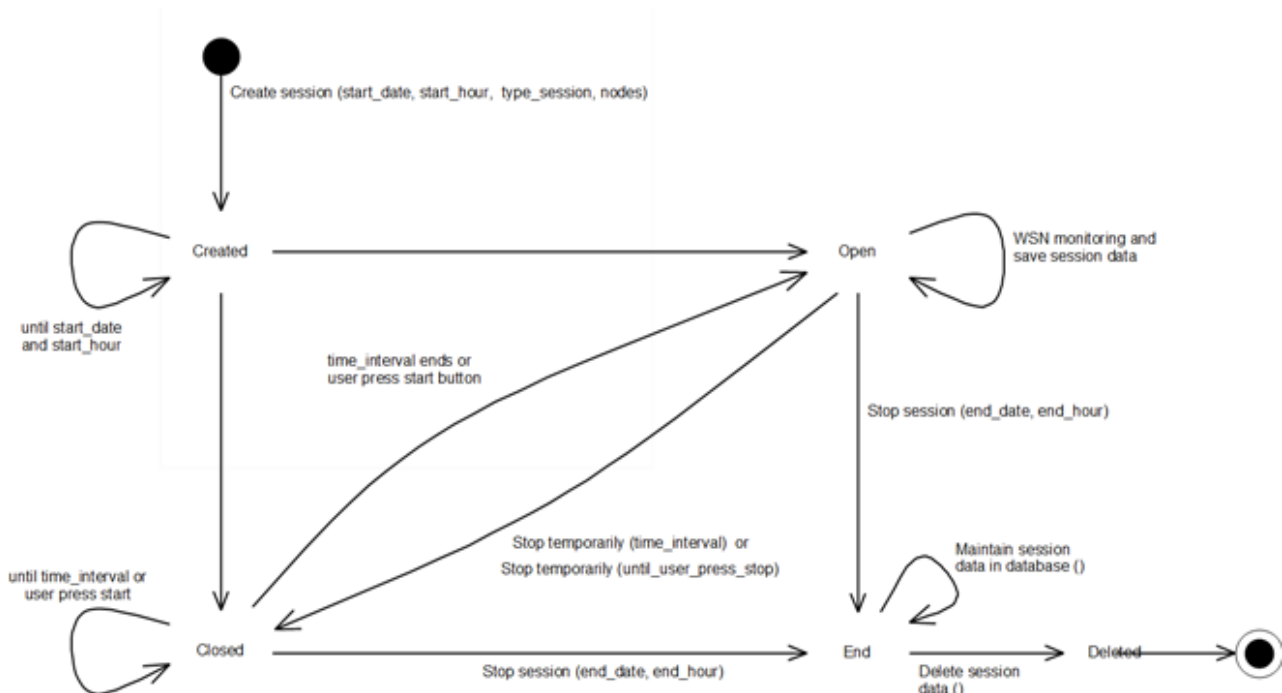


Figura III.19: Diagrama de transição de estados de uma sessão.

III.6. MODELO DE DADOS: DIAGRAMA ENTIDADE-RELACIONAMENTO

Um diagrama entidade-relacionamento - *E-R* - é uma representação gráfica que mostra como são efectuadas as relações entre as várias entidades da base de dados.

Uma entidade é um conjunto de objectos do mesmo tipo, acerca dos quais se pretende recolher e guardar informação. Ao efectuar isso, podemos ter essa informação organizada por categoria ou itens, designados por atributos.

Os atributos são os diferentes elementos ou itens de informação que caracterizam uma entidade. Os atributos de uma entidade são o que irá dar origem aos campos na tabela da base de dados. Existem dois tipos de atributos, os identificadores e os descritores [23]:

- Os identificadores permitem identificar inequivocamente todas as ocorrências das entidades, são chamados de chave primária.
- Os outros atributos são chamados de descritores, visto que apenas descrevem ou caracterizam as ocorrências das entidades.

Ao conectar as diferentes entidades, é formada uma relação entre as mesmas. Estas relações podem ser de diversos tipos [23]:

- Um para um (1:1): Uma ocorrência da uma entidade apenas corresponde a uma ocorrência da outra entidade.
- Um para muitos (1:N): A cada ocorrência de uma entidade, correspondem várias ocorrências de outra entidade, ou seja, a entidade que fica do lado do 1, tem a sua ocorrência a aparecer n vezes na outra entidade.
- Muitos para muitos (N:N): Neste tipo de relação, ambas as entidades contêm várias ocorrências na outra entidade. Quando ocorre este tipo de relação, é definida uma nova tabela, que actua como intermediária entre as duas entidades. As relações entre as entidades e a nova tabela, são de um para muitos, que já se encontra descrito em cima.

O modelo de dados surge pela análise efectuada aos requisitos definidos para a construção da ferramenta. Observando os requisitos, verificamos a necessidade de uma tabela que guarde as sessões colaborativas criadas pelos utilizadores. Esta tabela necessita de conter o nome e descrição da sessão, o horário

de monitorização, os nós da sessão e quais os dados que a sessão monitoriza. Desta forma, iríamos ter uma tabela como a exemplificada na Figura III.20.














session	
	idsession: INTEGER
	nomeSessao: VARCHAR(50)
	dataInicio: VARCHAR(50)
	horaInicio: VARCHAR(50)
	dataFim: VARCHAR(50)
	horaFim: VARCHAR(50)
	tipoSessao: ENUM('room','showcase','persona...')
	objectivo: VARCHAR(50)
	estado: ENUM('created','open','close','end')
	descricao: VARCHAR(50)
	monitorizar: BOOL
	nos: VARCHAR(50)
	dadosMonitorizar: VARCHAR(50)

Figura III.20: Modelo entidade-relação não normalizado.

Como esta base de dados não se encontra normalizada, é necessário proceder à sua normalização. Desta forma, foram criadas novas tabelas: para os dados a monitorizar, *dadosMonitorizar*, que irá incluir o nome do fenómeno a monitorizar. Na descrição dos nós, foi criada a tabela *nos*, que inclui os campos que são retirados da base de dados *WISE-MUSE*, a localização, o tipo de sensores, o que o sensor monitoriza, o tipo de alimentação e o nome do dispositivo. É também criada uma tabela para tratar as amostras recebidas pela rede, a *sessionvaloresrecebidos*, que guarda os valores das amostras e também a data e hora que a amostra foi recebida. A nova base de dados que contém todas estas alterações, encontra-se na Figura III.21.

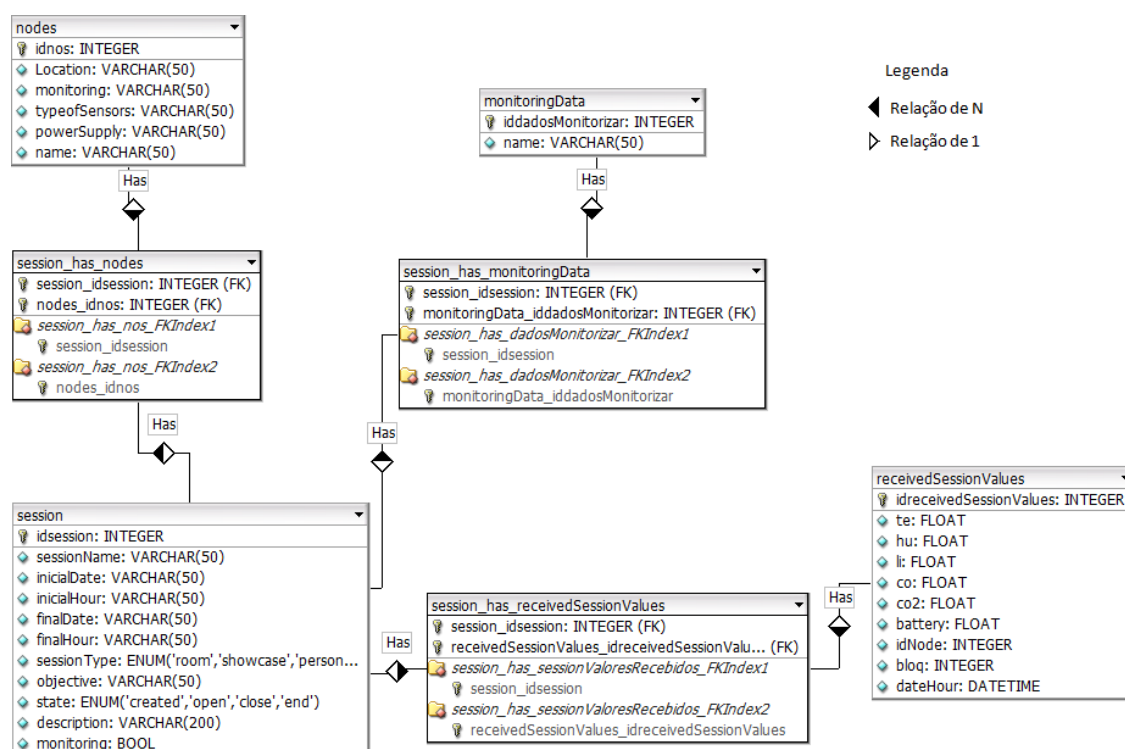


Figura III.21: Modelo de dados entidade-relação normalizado.

Neste modelo, apresentado na Figura III.21, uma sessão pode ter vários nós, e um nó pode estar em várias sessões diferentes, desta forma, temos uma relação $N:N$, o que leva a que a tabela *session_has_nodes* seja criada. Esta tabela é criada utilizando o atributo identificador, ou seja, a chave primária, de cada uma das entidades.

Uma sessão pode monitorizar diversos fenómenos, assim como um fenómeno pode estar presente em mais do que uma sessão. Desta forma, a relação das entidades *session* e *monitoring data* terá que ser $N:N$, o que leva a que seja criada uma nova tabela. Esta tabela, *session_has_monitoringData*, contém os identificadores de ambas as entidades.

Finalmente, a relação entre as entidades *receivedSessionValues* e *session* também é de $N:N$, visto que uma amostra ao ser recebida, pode ser de várias sessões, e uma sessão receberá mais do que uma amostra. Desta forma, é criada a tabela *session_has_receivedSessionValues*, que contém novamente os identificadores de ambas as entidades.

Desta forma, podemos verificar que o modelo de dados escolhido contém sete tabelas, sendo que três delas são utilizadas para fazer a ligação entre as restantes.

III.6.1.Descrição das entidades

A tabela nós contém todos os nós da rede, sendo que guarda a localização do nó, qual o tipo de sensor, o que esse sensor monitoriza, qual o tipo de alimentação e

qual o nome do mesmo. A *location* mostra, através de coordenadas X, Y e Z, a localização exacta de cada sensor. *Monitoring* mostra quais os fenómenos que cada sensor monitoriza, sendo mostrada uma abreviatura desse mesmo fenómeno:

- *Te*: Temperatura;
- *Hu*: Humidade;
- *Li*: Luz;
- *Co*: Monóxido de Carbono;
- *Co2*: Dióxido de Carbono;
- *Bl*: Bloqueador.

O campo *typeofSensors* guarda o tipo de cada sensor, que poderá ser coordenador, *router* ou *end-device*. O campo *powerSupply* mostra se o dispositivo é alimentado a bateria ou por energia eléctrica. Finalmente, o campo *name* guarda o identificador do dispositivo.

A tabela *monitoringData* contém quais os fenómenos que a rede monitoriza. Neste momento a rede monitoriza temperatura, humidade, luz, monóxido de carbono, dióxido de carbono e o estado das portas de emergência.

A tabela *session* guarda as informações das sessões criadas pelo utilizador. O nome da sessão é guardado no campo *sessionName*, a descrição da sessão fica guardada no campo *description*. O horário de monitorização da sessão fica guardado nos campos *initialDate*, *initialHour*, *finalDate* e *finalHour*. Quanto ao que a sessão monitoriza, esta pode ser de vários tipos:

- *Room*: Monitoriza os fenómenos temperatura, humidade e luz;
- *Showcase*: Monitoriza os fenómenos temperatura, humidade, luz, monóxido de carbono e dióxido de carbono.
- *Personalized*: Neste tipo de sessão o utilizador escolhe quais os fenómenos a monitorizar, de entre temperatura, humidade, luz, monóxido de carbono, dióxido de carbono e o estado dos bloqueadores das portas de emergência.

Finalmente, o campo *monitoring* guarda se a sessão está a monitorizar.

A tabela *receivedSessionValues* guarda os parâmetros das amostras recebidas pela rede, guardando também, a data e hora da mesma, que fica guardada no campo *dateHour*.

As restantes três tabelas são utilizadas na ligação das tabelas descritas anteriormente, para manter a base de dados normalizada. A tabela *session_has_nodes* mostra quais são os nós de cada sessão, a tabela *session_has_monitoringData* mostra quais são os fenómenos que cada sessão monitoriza. Finalmente, a tabela *session_has_receivedSessionValues*, guarda as amostras que foram recebidas por cada sessão.

III.7. CONCLUSÃO

A correcta identificação dos requisitos é essencial para a execução de qualquer projecto, para que o mesmo seja concluído com sucesso.

De forma a efectuar a integração com a plataforma *WISE-MUSE*, este sistema foi implementado para funcionar com o protocolo *ZigBee* e com os dispositivos *XBee*.

Ao observar os requisitos do sistema, pode-se concluir que o mesmo divide-se em duas partes, uma que permite comunicar com a rede e outra que permita criar e gerir as sessões colaborativas. Na comunicação com a rede é necessário efectuar a recolha das amostras da rede e também permitir que o utilizador comunique com a mesma. Nas sessões colaborativas, é necessário garantir que o utilizador consegue criá-las, manipulá-las e visualizar as amostras recolhidas pelas mesmas.

Para corresponder aos requisitos, o modelo de dados a utilizar será constituído por sete tabelas, para conseguir dar resposta a todas as necessidades dos *stakeholders*. Desta forma, o sistema permite que as sessões sejam constituídas por vários dispositivos que monitorizam diversos fenómenos.

No próximo capítulo será descrita a implementação do sistema *WISE-MANager*, assim como os testes de cobertura do sinal efectuados no Museu da Baleia.

IV. IMPLEMENTAÇÃO

IV.1. INTRODUÇÃO

Neste capítulo são abordados dois aspectos importantes implementados neste projecto de mestrado: (i) um estudo de propagação realizado no Museu da Baleia, situado no Caniçal, e (ii) o sistema *WISE-MANager*. Para o estudo de propagação serão apresentados o equipamento e o *software* utilizado nas medições e testes e a ferramenta *RSSI Level* que obtém o valor *RSSI* dos módulos *XBee*.

Por outra parte, serão apresentados os dois componentes principais do sistema *WISE-MANager*: (i) *Collaborative Session Management*; e, (ii) *WSN Management*.

O primeiro componente permite a criação e monitorização de sessões colaborativas, enquanto o segundo gere a rede de sensores sem fios. Em seguida, será apresentado o *hardware* e *software* utilizado e o processo de implementação.

IV.2. HARDWARE E SOFTWARE UTILIZADO NA IMPLEMENTAÇÃO

Os módulos *XBee*, fabricados pela *MaxStream*, foram os módulos utilizados no estudo de propagação do sinal e na programação do sistema *WISE-MANager*, sendo que incluem todo o *hardware* e a lógica necessária para implementar uma rede *ZigBee*. A *MaxStream* fabrica duas versões do *XBee*: *XBee* e *XBee-Pro*, que se diferenciam na potência de saída e no alcance máximo do sinal, como podemos observar na tabela IV.1. Ambas as versões estão disponíveis com três tipos de antena: chicote (*whip*), *chip* e conector para antena externa, que estão exemplificados na Figura IV.1 [26]:

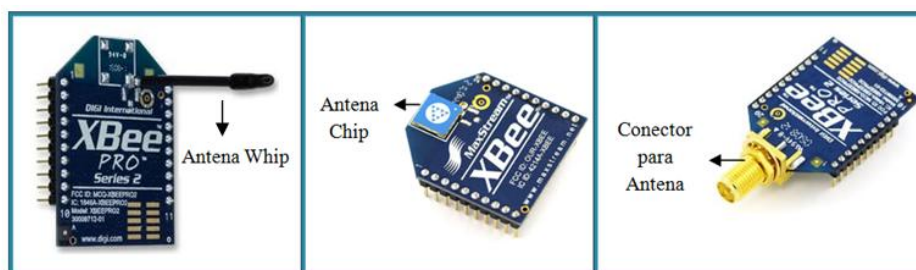


Figura IV.1: Tipos de *XBee*.

Tabela IV.1: Características dos *XBee*.

	<i>XBee Series 2 pro</i>	<i>XBee Series 2</i>
Frequência de operação	2.4 GHz	
Alcance em ambiente <i>indoor</i>	100 m	40 m
Potência de Transmissão	10mW	2mW
Sensibilidade do receptor	-102 dBm	-96 dBm

O equipamento principal utilizado na realização do estudo de propagação da rede e na programação do *WISE-MANager* foi o seguinte:

- **Coordenador:** Placa série USB *Maxstream*, *XBee series 2 pro*, cabo USB e PC portátil;
- **End-device:** Placa série USB *Maxstream*, *XBee series 2 pro*, cabo USB e PC portátil;
- **Bloqueador:** Placa série USB *Maxstream* *XBee series 2* com conector para antena, cabo USB e PC portátil.

Para configurar os dispositivos de teste utilizou-se o *software* X-CTU da *MaxStream*, cuja configuração encontra-se descrita de seguida:

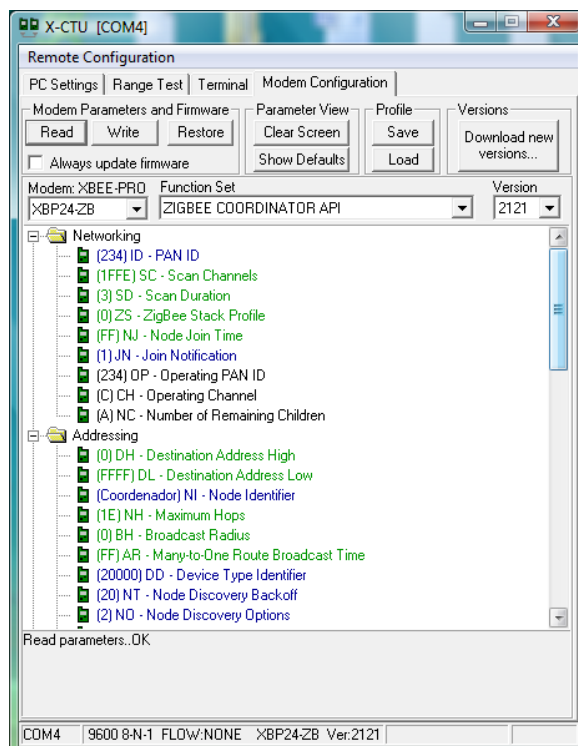


Figura IV.2: Configuração do coordenador da rede.

Como podemos observar na Figura IV.2, o coordenador permite definir o ID da rede, o *PAN ID*, o canal de comunicação, *CH*, entre outras opções. Na Figura IV.3 encontra-se a configuração do router, de forma a ser utilizado nos testes.

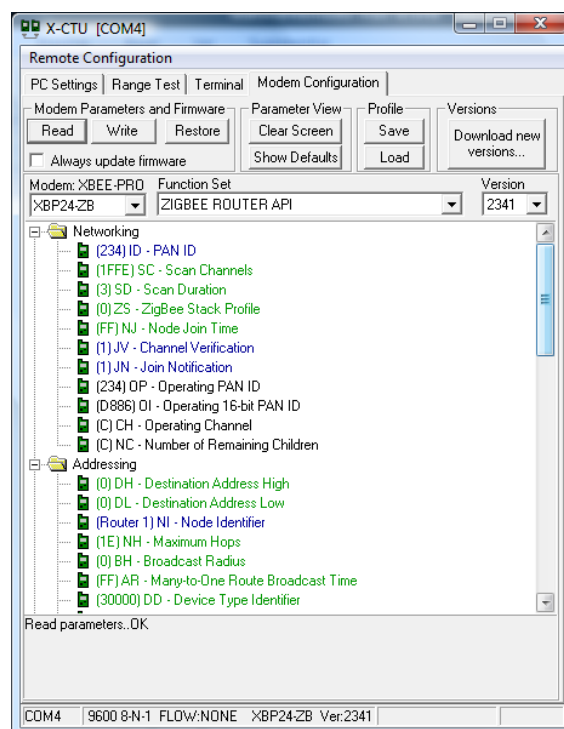


Figura IV.3: Configuração do router utilizado nos testes.

Para a realização dos testes foi efectuado um programa utilizando a *API XBee*, disponibilizada em <http://code.google.com/p/XBee-API/>, para efectuar a comunicação com a rede, e construída uma interface que permitisse escolher qual a porta COM a utilizar e qual o endereço do *XBee* para verificar o *RSSI*. Este programa encontra-se descrito na secção IV.3.2.

IV.2.1. Software

O *software* utilizado para criar a aplicação utilizada no estudo da propagação do sinal no Museu da Baleia e para criar o *WISE-MANager* foi o *Eclipse*, utilizando a linguagem de programação *Java*. O *Eclipse* é um ambiente de desenvolvimento de *software* capaz de utilizar várias linguagens, permitindo a utilização de um ambiente de desenvolvimento integrado (*IDE*) e a utilização de um sistema de *Plug-in* extensível. Este *IDE* foi criado em, e para *Java*, mas permite desenvolver aplicações em outras linguagens, por exemplo em *C*, *C++*, *PHP*, entre outras. Este *software* é livre e *open source*, podendo ser obtido em <http://www.eclipse.org/downloads/> [27].

Outro *software* utilizado neste projecto de Mestrado foi o *X-CTU*, que foi utilizado para configurar os dispositivos *XBee* utilizados nos testes. Este *software*, criado pela *Digi*, permite a configuração e testes dos modems rádio *MaxStream*, e pode ser obtido, de forma gratuita, em <http://www.digi.com/support/productdetl.jsp?pid=3352&osvid=57&s=316&tp=5&tp2=0>. Com a utilização deste *software*, escolhemos qual o modo de comunicação, *AT* ou *API*, sendo que o modo utilizado no desenvolvimento deste projecto é o *API*, visto que o *AT* apenas permite a criação e envio de comandos simples, não permitindo a utilização dos comandos que foram utilizados e que serão descritos posteriormente.

IV.2.2. API's utilizadas

Uma *API* – *Application Programming Interface* –, ou Interface de Programação Gráfica, é uma interface que é implementada por aplicações, bibliotecas e sistemas operativos de forma a permitir o acesso aos seus serviços. Pode incluir especificações tais como rotinas, estruturas de dados ou classes, de forma a permitir uma comunicação entre o utilizador da *API* e o construtor da mesma [28].

IV.2.2.1. API XBee

Esta *API* foi desenvolvida por *Andrew Rapp* e é disponibilizada em <http://code.google.com/p/XBee-API/>. A *API XBee* foi criada em Abril de 2008 e actualmente encontra-se na versão 0.5.5, que foi lançada em Maio de 2009.

Esta *API* *java* permite a comunicação com os módulos *XBee/XBee-Pro series 1* e *series 2 (ZNet 2.5 e ZB/ZigBee Pro)* no modo *API*.

IV.2.2.2.SWT

A *API SWT – Standard Widget Toolkit* - foi originalmente desenvolvida pela *IBM*, sendo que neste momento é utilizada pelo *IDE Eclipse* e mantida pela Fundação Eclipse.

Foi escrita em, e para Java, permitindo a utilização de elementos *GUI – Graphical User Interface* – como botões, *labels*, campos de texto, *combo box*, entre outros. Esta *API* permite uma total liberdade na criação da interface, sendo possível escolher a localização exacta de cada objecto. Esta *API* foi licenciada pela *Eclipse Public License* sendo por isso *open source* [29].

Em seguida mostraremos o estudo de propagação do sinal efectuado no Museu da Baleia.

IV.3. ESTUDO DE PROPAGAÇÃO DO SINAL

Para a validação do projecto *WISE-MUSE* e do *WISE-MANager* será instalada uma WSN no Museu da Baleia. Para obter o melhor *deployment* da rede, ou seja, determinar qual a melhor posição para a instalação dos dispositivos da WSN, foi necessário efectuar diversos testes, para observar como o sinal de propagava no interior do museu.

Este estudo de propagação do sinal consistiu em medir o valor de *RSSI* - *Received Signal Strenght Indication* - nas salas do museu a monitorizar. Os dispositivos foram colocados aproximadamente a uma altura de dois metros e juntos à parede, para conseguirmos simular melhor o comportamento dos *XBee*.

O sistema de monitorização que será instalado no Museu da Baleia é baseado numa rede heterogénea, utilizando dois tipos de dispositivos. Um é um nó sensor que colecta os dados ambientais tais como a temperatura, humidade, luminosidade e gases poluentes (monóxido de carbono, CO e dióxido de carbono, CO₂). O outro tipo de dispositivos, localizado dentro dos bloqueadores das portas de emergência, detecta o estado das mesmas. As portas de emergência podem estar nos seguintes estados:

Tabela IV.2: Estados das portas de emergência.

Estado	Descrição	Cor
Aberto	A porta encontra-se aberta.	Amarelo
Fechado	A porta encontra-se fechada.	Vermelho
Emergência	Quando a porta é aberta sem autorização.	Amarelo a piscar

IV.3.1.Descrição do Cenário

O Museu da Baleia encontra-se localizado na vila do Caniçal e é um testemunho de toda a história da caça à baleia e das actividades a ela associadas [32].

O Museu comporta quatro andares com diversas salas em cada um deles. No piso 0, pretende-se monitorizar a sala de cetáceos, a sala caça da baleia, o átrio e a sala do segurança no parque de estacionamento. Além destas salas, é necessário monitorizar o estado de nove portas de emergência. A sala dos cetáceos é a maior sala a monitorizar, ocupando três andares (o piso 0, 1 e 2), e contendo diversos obstáculos: duas baleias de grandes dimensões no centro da sala; um submarino e vários golfinhos ao longo da sala, suspensos pelo tecto. Na sala projecção de vídeo existem várias cadeiras dispostas em forma de anfiteatro e um ecrã de grandes dimensões. A sala de caça da baleia tem vários quadros, uma baleia e um barco,

ambos juntos à parede existindo também várias vitrinas. O átrio é uma zona de passagem, que se encontra decorada com vários quadros. Na sala do segurança não existem obstáculos que impeçam a passagem do sinal.

No piso 1 pretende-se monitorizar o auditório e a sala de exposições temporárias. Neste andar tem-se cinco bloqueadores de portas. O auditório tem as cadeiras do tipo anfiteatro, mas não possui obstáculos que bloqueiem o sinal. Em relação à sala de exposições temporárias, encontrava-se vazia na realização do teste.

No piso 2 encontra-se a sala didáctica A e a sala didáctica B que não possuem obstáculos suspensos pelo tecto que possam obstruir o sinal. Neste andar existem três bloqueadores de portas.

No piso -1 vamos monitorizar o depósito de peças museológicas, o depósito de material biológico, a câmara frigorífica e a câmara congeladora, sendo que neste piso existem dois bloqueadores.

Concluindo, o cenário utilizado é uma rede heterogénea composta por nós sensores que monitorizam a temperatura, humidade, luminosidade e gases poluentes (monóxido de carbono, CO e dióxido de carbono, CO₂) e por bloqueadores que detectam o estado das portas de emergência.

IV.3.2. Software utilizado

Uma das contribuições deste projecto é o *RSSI Level*, uma aplicação desenvolvida em Java que permite visualizar o valor de *RSSI* dos vários dispositivos da rede. Esta aplicação surgiu com a necessidade de obter os valores de *RSSI* de forma fácil e rápida. É utilizada a *API XBee* para Java, descrita na secção IV.2.2.1, para efectuar a comunicação com a rede *XBee*.

Na Figura IV.4 encontra-se a interface inicial do programa. O utilizador necessita de escolher qual a porta do coordenador, de forma a enviar o comando para saber o nível de sinal. Neste caso, a porta escolhida foi a COM6.

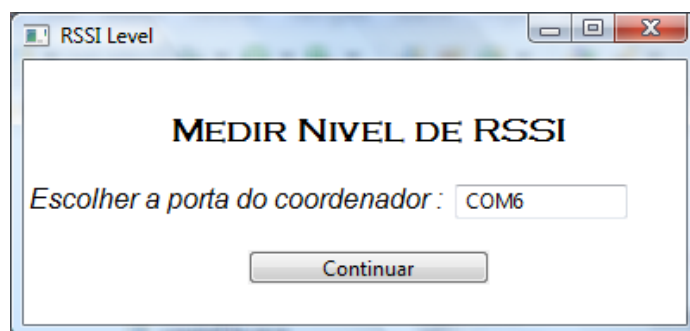


Figura IV.4: Escolher porta do coordenador no programa *RSSI Level*.

Depois de escolher a porta *COM*, e clicar em continuar, é apresentado o resto da interface, onde são preenchidas as caixas de texto com o endereço de 64 bits do dispositivo *XBee* que queremos saber o sinal, como pode ser observado na Figura IV.5. Seguidamente a preencher o endereço do dispositivo, apenas é necessário clicar em “Ver Nível de Sinal”, que o valor de *RSSI* do dispositivo será mostrado. Isto está

apresentado na Figura IV.6, sendo que o dispositivo escolhido foi o `00 13 a2 00 40 2d d7 ca`, onde podemos verificar qual o nível de *RSSI* do dispositivo.



Figura IV.5: Escolher o endereço para saber o nível de sinal.



Figura IV.6: Sinal *RSSI* do endereço `00 13 a2 00 40 2d d7 ca`.

IV.3.2.1. Comunicação entre a Plataforma e a Rede

O valor de *RSSI* que se encontra na Figura IV.6 é obtido com o envio do comando *DB* para o dispositivo em questão. Com a utilização da *API XBee*, primeiro é necessário abrir a porta de conexão, entre a plataforma e o coordenador, que está exemplificado na Figura IV.7, onde *portacom* é a porta escolhida pelo utilizador.

```
XBee xbee = new XBee();
xbee.open(portacom, 9600);
```

Figura IV.7: Criar a conexão entre o *XBee* e a plataforma.

Posteriormente a abrir a conexão, apenas é necessário efectuar o envio do comando, e guardar o pacote que é recebido com o valor de *RSSI*. Na Figura IV.8, encontra-se a forma como o comando é criado, utilizando o endereço *addr64* escolhido pelo utilizador na Figura IV.5, e o comando a enviar para esse endereço, neste caso o *DB*, que retorna o valor de *RSSI* do dispositivo. Para receber esse valor, é criada uma comunicação síncrona, descrita na Figura IV.9, que envia a *request* criada

na Figura IV.8. Em seguida, é recebido o pacote com a resposta, que fica guardada na variável *response*. De forma a descobrir qual o valor de *RSSI* apenas é necessário recorrer ao método *getValue()* da variável *response*.

```
ZNetRemoteAtRequest request = new ZNetRemoteAtRequest(addr64, "DB");
```

Figura IV.8: Criação do comando DB.

```
ZNetRemoteAtResponse response = (ZNetRemoteAtResponse) xbee.sendSynchronous(request, 10000)
```

Figura IV.9: Envio do comando DB.

IV.3.3.Descrição do Estudo de Propagação

Foram efectuados oito testes no Museu da Baleia, devido à grande extensão do mesmo. Na realização dos testes utilizou-se dois dispositivos, um configurado como coordenador que envia o comando *DB* para o outro dispositivo (que pode ser configurado como *router* ou *end-device*), e este envia o seu valor de *RSSI*. Só utilizamos dois dispositivos porque a introdução de mais dispositivos levava a que o valor de *RSSI* recebido não fosse o real, ou seja, esse valor recebido seria apenas o valor do último router para o coordenador, e não do dispositivo que queríamos testar.

A Figura IV.10 mostra o piso 0 e os valores *RSSI* obtidos:



Figura IV.10: Medições no piso 0.

Os restantes testes e a topologia da rede estão descritos com maior detalhe no anexo A.

IV.4. WISE-MANAGER

Outra contribuição deste projecto de Mestrado é o sistema *WISE-MANager*. Este sistema foi desenvolvido em Java, e contém 23 classes, que podemos observar no respectivo diagrama da figura IV.11:

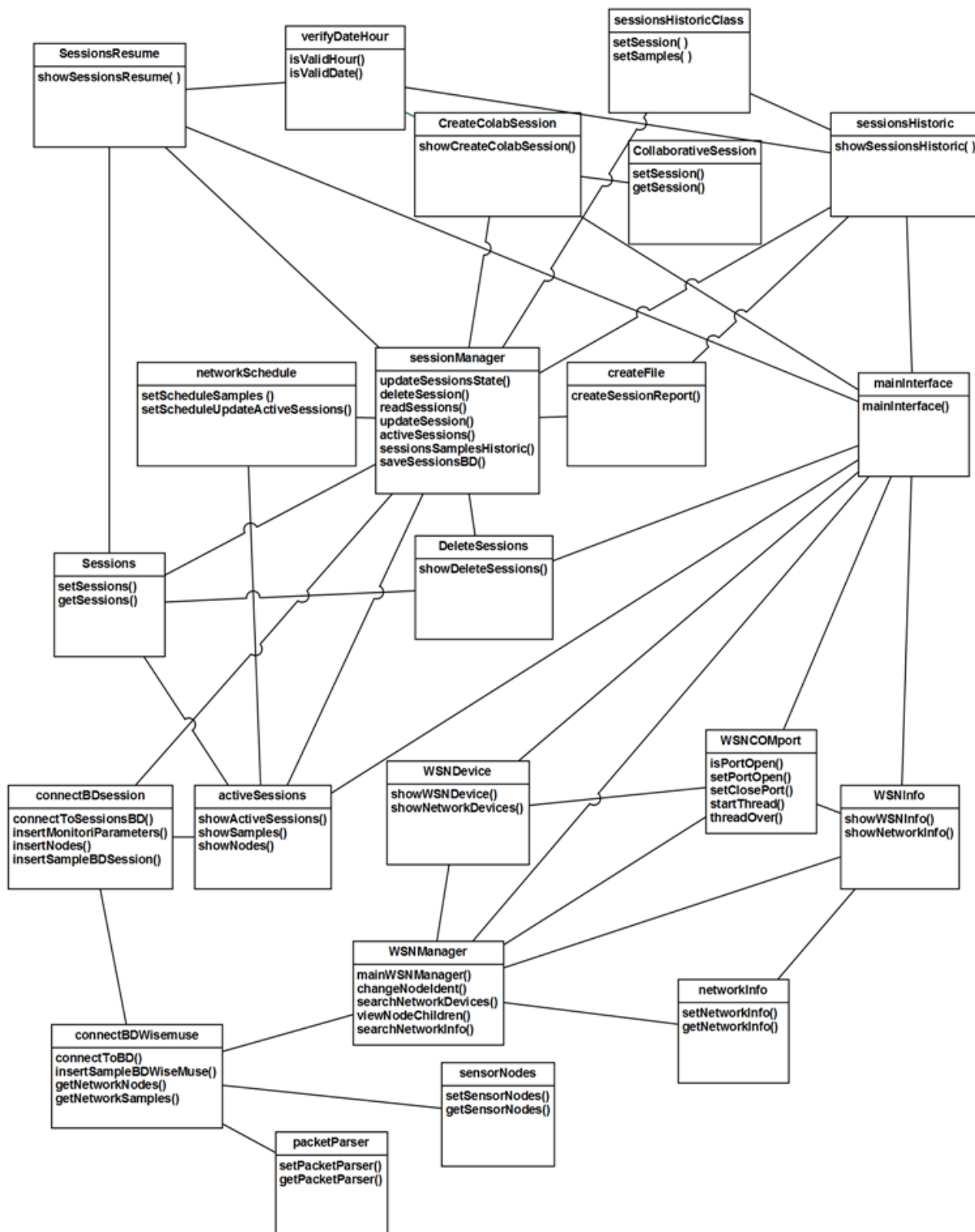


Figura IV.11: Diagrama de classes do sistema *WISE-MANager*.

De seguida, são descritas todas as classes do sistema, assim como as operações que as mesmas executam:

A classe *mainInterface* é a classe que é carregada ao iniciar o programa. É através da sua execução que são criados todos os menus e interfaces para serem utilizados pelo utilizador.

A classe *Sessions* é utilizada por diversas classes, visto que é a classe que guarda todas as sessões que são inseridas na base de dados. Esta classe contém diversos métodos que permitem inserir os dados retirados da base de dados, como por exemplo, *setNome()* e *setDescricao()*, e também permite aceder a esses atributos através de *getNome()* e *getDescricao()*, entre outros métodos.

Na criação da sessão colaborativa, são utilizadas as classes *CreateColabSession*, *verifyDateHour* e *CollaborativeSession*:

A classe *CreateColabSession* é responsável por apresentar a interface da criação de novas sessões colaborativas ao utilizador, através do método *showCreateColabSession*.

A classe *CollaborativeSession* é utilizada para guardar os parâmetros escolhidos pelo utilizador na criação da sessão colaborativa, sendo contém diversos atributos e métodos para esta tarefa.

A classe *verifyDateHour* é utilizada para validar os dados inseridos pelo utilizador, mais precisamente se a data e hora inserida estão nos formatos correctos, através da execução dos métodos *isValidDate()* e *isValidHour()*.

A classe *connectBDSession* é utilizada para efectuar a conexão com a base de dados das sessões, utilizando o método *connectToSessionsDB()*. Esta classe é utilizada para inserir os parâmetros a monitorizar, com o método *insertMonitorParameters()* e para inserir as amostras na base de dados, executando o método *insertSampleBDSession()*.

A classe *SessionManager* possui todos os métodos necessários para: apagar as sessões da base de dados (*deleteSession()*), para efectuar a leitura das sessões da base de dados (*readSessions()*), verificar quais as sessões activas (*activeSessions()*), recolher o histórico da sessão (*sessionsSamplesHistoric()*) e guardar as sessões, na base de dados, criadas pelo utilizador (*saveSessionsDB()*).

A classe *networkSchedule*, utilizando a classe *TimerTask*, cria dois *timers* que são utilizados para verificar a chegada de novas amostras da rede (método *setScheduleSamples()*) e para actualizar a interface que mostra as sessões activas (método *setScheduleUpdateActiveSessions()*). A verificação de novas amostras é feita de 10 em 10 minutos, enquanto que a actualização das sessões é de minuto a minuto.

A classe *sessionsResume* é utilizada para apresentar a interface que mostra todas as sessões da base de dados, e permite alterá-las.

A classe *sessionsHistoric* fornece a interface que permite ao utilizador gerar o histórico das sessões, utilizando o método *showSessionsHistoric()*. São guardados na classe *sessionsHistoricClass* os valores retirados da base de dados para efectuar o relatório pedido pelo utilizador. Esta classe fornece os métodos para guardar e aceder aos dados necessários para a criação do histórico da sessão, ou seja, aos parâmetros da sessão, assim como às amostras recolhidas por esta sessão. Na classe

createFile, é criado o ficheiro *MS Word* com as amostras obtidas pela sessão e intervalo de tempo escolhido pelo utilizador, utilizando o método *createSessionReport()*.

A classe *deleteSessions* apresenta a interface para apagar sessões da base de dados, mostrando as informações das sessões e um botão para apagar a mesma, usando para isto o método *showDeleteSession()*.

A classe *connectBDWisemuse* efectua a comunicação com esta base de dados, através do método *connectToDB()*. Ao utilizar o método *getNetworkNodes()*, recolhe as informações dos nós que se encontram na base de dados *WISE-MUSE*, guardando-os na classe *sensorNodes*. O método *getNetworkSamples()* permite retirar as amostras desta base de dados, sendo depois guardados na base de dados das sessões colaborativas, utilizando para isso a classe *packetParser* que permite retirar os dados das amostras da base de dados *WISE-MUSE*.

A classe *WSNManager* é responsável por todo o tratamento de dados da comunicação entre a plataforma e a rede de sensores sem fios. Para isto, é criada uma *Thread*, que corre paralelamente ao programa, tratando da recepção dos pacotes assíncronos da rede, como por exemplo, as amostras enviadas pela *WSN*. Depois de recolhidas, estas amostras são inseridas na base de dados *WISE-MUSE*. Esta classe também é responsável pelo envio dos comandos síncronos para a rede, com a utilização dos métodos *searchNetworkInfo()*, *viewNetworkChildren()* e *changeNodeIdent()*.

A classe *WSNCOMport* trata da abertura, e fecho, da porta de comunicação entre o sistema *WISE-MANager* e a *WSN*, através dos métodos *setPortOpen()* e *setClosePort()*.

A classe *WSNDevice* é responsável por mostrar ao utilizador os dispositivos da rede, utilizando os métodos *showWSNDevice()* e *showNetworkDevices()*, assim como permitir que o mesmo altere o nome dos dispositivos da rede.

A classe *WSNInfo* mostra ao utilizador as informações básicas da rede, utilizando para isso a classe *networkInfo*, que guarda as informações recolhidas da rede, através do método *showWSNInfo()*.

Este sistema tem como objectivos efectuar a monitorização da rede, permitir a gestão da mesma e efectuar a criação e gestão de sessões colaborativas. Na Figura IV.12 encontra-se a interface inicial deste sistema.

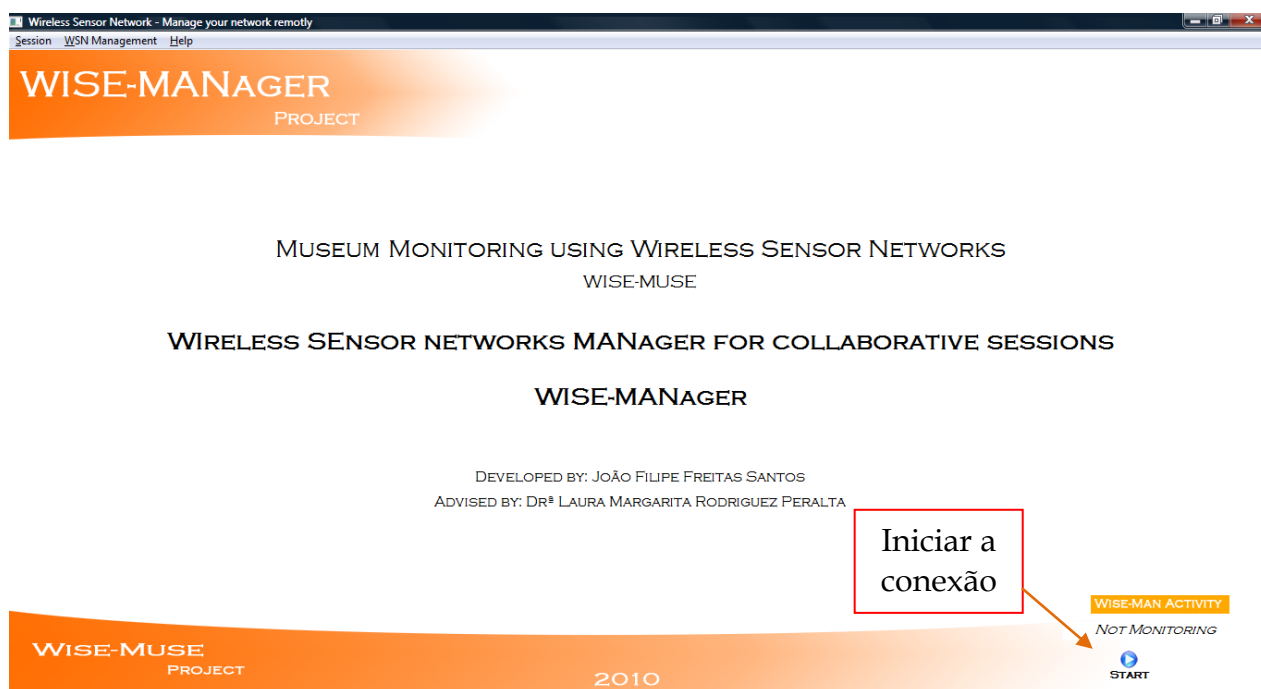


Figura IV.12: Interface inicial da aplicação *WISE-MANager*.

Como podemos observar pela Figura IV.12, no menu superior, existem as opções “*Session*”, “*WSN Management*” e “*Help*”. No canto inferior direito, existe um botão que permite iniciar a conexão entre a rede e o sistema. Nas próximas secções serão apresentados estes menus, assim como a conexão com a rede *XBee*.

IV.4.1.Collaborative Session Management

Neste menu o utilizador pode criar sessões colaborativas, visualizar quais são as sessões activas, visualizar e alterar os dados das sessões colaborativas, exportar os dados para *MS Word* e apagar a sessão da base de dados.

IV.4.1.1.Criação da Sessão Colaborativa

Para criar a sessão colaborativa é necessário preencher os vários parâmetros da mesma, que serão descritos de seguida:

Identificador da Sessão:

Id escolhido pelo utilizador para identificar a sessão que está a criar, sendo que esse identificador não pode ser um que já exista numa outra sessão criada previamente.

Descrição:

O utilizador preenche uma pequena descrição da sessão, que pode ser visualizada no resumo das sessões, ou ao observar as sessões activas.

Monitorização:

O utilizador terá que escolher o horário de monitorização da sessão, definindo qual a data e hora em que a sessão irá recolher valores. Desta forma, temos a restrição que a data de início terá que ser maior que a data actual e a data fim terá que ser posterior à data de início. Quanto às horas, a hora de fim da monitorização terá que ser posterior à data de início de monitorização.

Dispositivos da rede:

O utilizador terá que escolher quais os dispositivos que irão fazer parte da sessão, para isso, terá algumas informações sobre cada nó a escolher:

- Identificador: O *Id* que é dado ao nó de modo a facilitar a sua identificação, por exemplo, Nó 1.
- Localização: São disponibilizadas as coordenadas de cada dispositivo, a X, Y e Z. Desta forma, o utilizador sabe a posição real de esse dispositivo.
- Tipo de Sensor: Um sensor pode ser de vários tipos: Temperatura, Humidade e Luz; Temperatura, Humidade, Luz, Monóxido de Carbono e Dióxido de Carbono; ou Bloqueador. Assim, o utilizador ao escolher um dispositivo, sabe o que este monitoriza.
- Tipo de dispositivo: Existem três tipos de dispositivos, coordenador, *router* ou *end-device*. Com esta informação, o utilizador sabe qual o tipo de dispositivo que está a escolher para formar a sessão.

Escolher nós para a sessão:

Depois de observar quais os dispositivos da rede, para escolher algum é necessário seleccionar o mesmo clicando na caixa que se encontra antes do nó para colocar um *tick* no mesmo, como podemos observar na Figura IV.13.

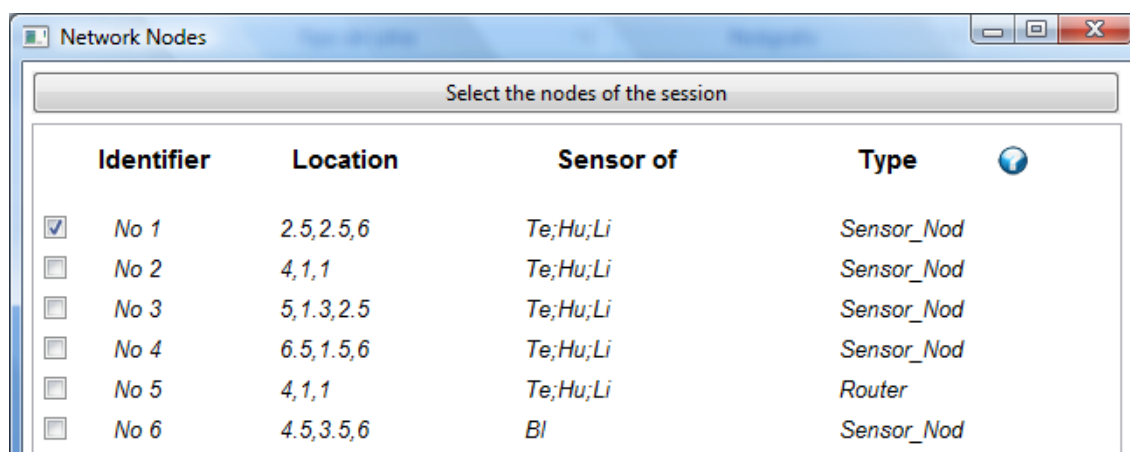


Figura IV.13: Escolher os nós para a sessão.

Monitorização:

Existem vários tipos de sessões dependendo da necessidade do utilizador e da quantidade de sensores dos nós escolhidos:

- Vitrina (*Showcase*): Irá monitorizar os parâmetros temperatura, humidade e luz.
- Sala (*Room*): Irá monitorizar os parâmetros temperatura, humidade, luz, monóxido e dióxido de carbono.
- Personalizado (*Personalized*): O utilizador pode escolher entre os vários parâmetros descritos em cima e pode escolher monitorizar as portas de emergência.

Na Figura IV.14, encontra-se a criação de uma sessão colaborativa, utilizando o sistema *WISE-MANager*. Ao preencher os dados, o utilizador tem sempre acesso a uma pequena ajuda, que pode ser obtida ao clicar no botão “?”, que podemos observar na Figura IV.14. Ao clicar neste botão, será mostrada uma nova janela com uma pequena ajuda para preencher o parâmetro em questão, como está exemplificado na Figura IV.15.

Figura IV.14: Interface de “criação de sessão colaborativa”.

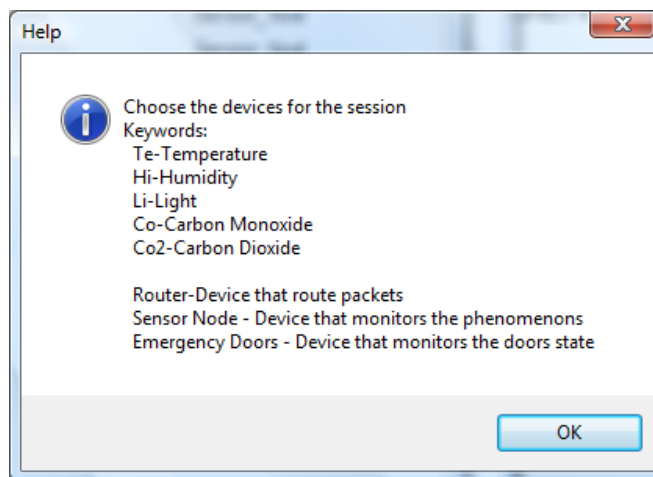


Figura IV.15: Janela da ajuda na escolha dos nós da sessão.

Na Figura IV.14, ao de clicar em “*Create Session*”, e se todos os dados estiverem preenchidos de forma correcta, é apresentada uma mensagem de confirmação que a sessão foi inserida com sucesso na base de dados.

IV.4.1.2. Visualizar Sessões Activas

A classe *sessionManager* é responsável pela correcta passagem das sessões entre os vários estados (*create*, *open*, *close* e *end*), de acordo com o horário de monitorização de cada uma, ou por intervenção do utilizador. Nesta janela, na Figura IV.16, são apresentadas as sessões que se encontram no estado *open*:



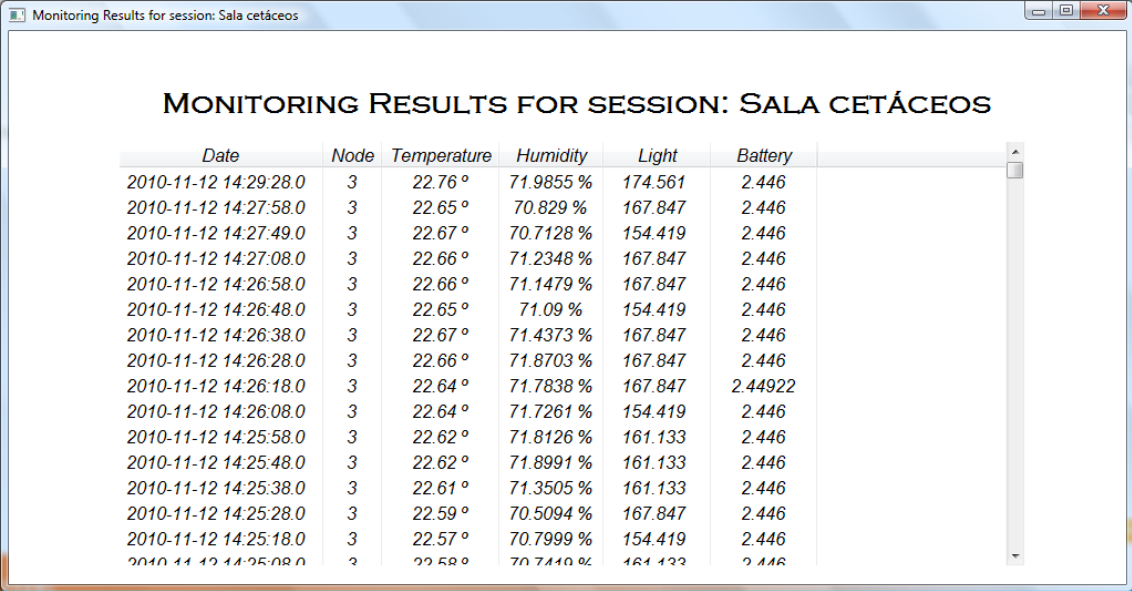
Figura IV.16: Listagem de todas as sessões activas.

Na Figura IV.16, observamos que neste momento encontra-se uma sessão *open*, ou seja, encontra-se a monitorizar. Ao verificar quais as sessões activas, é mostrado ao utilizador parâmetros como o nome e descrição da sessão, qual o horário que a sessão encontra-se no estado *open* e qual o tipo da sessão, ou seja, os fenómenos que a sessão monitoriza.

O utilizador também pode visualizar quais os dados monitorizados pela sessão e quais os nós que compõem a mesma. Estas funcionalidades serão descritas de seguida.

Dados Monitorizados

Para visualizar os dados monitorizados por esta sessão, é necessário escolher esta opção na Figura IV.16. Como podemos observar na Figura IV.17, são mostrados os parâmetros monitorizados assim como qual a data e hora que cada amostra foi recebida.



Date	Node	Temperature	Humidity	Light	Battery
2010-11-12 14:29:28.0	3	22.76 °	71.9855 %	174.561	2.446
2010-11-12 14:27:58.0	3	22.65 °	70.829 %	167.847	2.446
2010-11-12 14:27:49.0	3	22.67 °	70.7128 %	154.419	2.446
2010-11-12 14:27:08.0	3	22.66 °	71.2348 %	167.847	2.446
2010-11-12 14:26:58.0	3	22.66 °	71.1479 %	167.847	2.446
2010-11-12 14:26:48.0	3	22.65 °	71.09 %	154.419	2.446
2010-11-12 14:26:38.0	3	22.67 °	71.4373 %	167.847	2.446
2010-11-12 14:26:28.0	3	22.66 °	71.8703 %	167.847	2.446
2010-11-12 14:26:18.0	3	22.64 °	71.7838 %	167.847	2.44922
2010-11-12 14:26:08.0	3	22.64 °	71.7261 %	154.419	2.446
2010-11-12 14:25:58.0	3	22.62 °	71.8126 %	161.133	2.446
2010-11-12 14:25:48.0	3	22.62 °	71.8991 %	161.133	2.446
2010-11-12 14:25:38.0	3	22.61 °	71.3505 %	161.133	2.446
2010-11-12 14:25:28.0	3	22.59 °	70.5094 %	167.847	2.446
2010-11-12 14:25:18.0	3	22.57 °	70.7999 %	154.419	2.446
2010-11-12 14:25:08.0	3	22.58 °	70.7410 %	161.133	2.446

Figura IV.17: Dados monitorizados pela sessão Sala Cetáceos.

Nós da Sessão

Para visualizar quais são os nós que fazem parte da sessão, apenas é necessário escolher essa opção na janela que se encontra na Figura IV.16. Ao efectuar isto, é apresentada a Figura IV.18

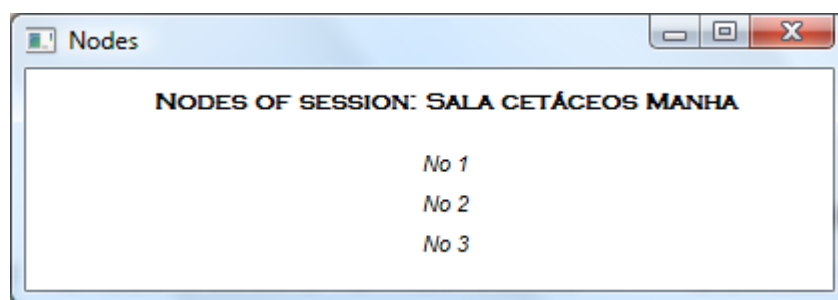


Figura IV.18: Nós da sessão “Sala Cetáceos Manhã”.

Nesta Figura observamos que os nós 1, 2 e 3 fazem parte da sessão “Sala Cetáceos Manhã”.

IV.4.1.3.Listagem das Sessões

Na Figura IV.19 é apresentada a lista de todas as sessões. Para obter esta lista, é efectuada uma consulta à base de dados. Em cada página são apresentadas quinze sessões, sendo que para visualizar outras sessões é necessário mudar de página na parte inferior da mesma.

Além de visualizar os dados das sessões, o utilizador pode alterar os mesmos, ou seja, alterar o identificador da sessão e a descrição da mesma. Também permite alterar o horário de monitorização, alterando a data e hora de início e de fim da monitorização. O tipo de sessão também pode ser alterado, entre *showcase*, *room* e *personalized*. Ao alterar o tipo de sessão, o utilizador altera os fenómenos que a sessão está a monitorizar.

O utilizador pode decidir começar, ou parar instantaneamente uma sessão. Isto poderá ser útil quando os fenómenos que a sessão está a monitorizar estão adulterados, permitindo dessa forma não guardar dados errados. Para começar a monitorização, é necessário carregar em “Start Session” e para parar a sessão carrega em “Stop Session”. A opção inicial para cada sessão será a “Automatic”, em que cada sessão irá monitorizar no seu horário pré-definido.

O utilizador pode observar as amostras recolhidas por cada sessão pressionando o botão “Session Samples”, que por sua vez abre uma janela semelhante à da Figura IV.17.

Depois de efectuar as alterações, o utilizador carrega no botão *Confirm Changes*, que irá actualizar os dados na base de dados.

WISE-MANAGER PROJECT

RESUME OF COLLABORATIVE SESSIONS

Session name	Description of the session	Monitoring Schedule	Session Type	State	Actual Session State
Temperatura, humidade	Ver a temperatura, humidade na sala	10/10/2010 at 08:00 to 10/10/2011 at 23:00	Personalized	close	Start Session Stop Session Automatic
Insert Session Name	Insert Description of session	10/10/2010 at 08:00 to 10/10/2011 at 12:00	Showcase	close	Start Session Stop Session Automatic
Teste 1	Sessão que monitoriza das 9:15 as 9:	15/11/2010 at 09:15 to 15/11/2010 at 09:20	Showcase	end	Start Session Stop Session Automatic
Teste 2 - Sessao 1	Sessao monitoriza a temperatura do	15/11/2010 at 16:15 to 15/11/2010 at 16:19	Personalized	end	Start Session Stop Session Automatic
Teste 2 - Sessao 2	Sessao monitoriza a humidade do no	15/11/2010 at 16:15 to 15/11/2010 at 16:19	Personalized	end	Start Session Stop Session Automatic
Teste 3	Sessao com dois nós a monitorizar te	15/11/2010 at 16:32 to 15/11/2010 at 16:36	Showcase	end	Start Session Stop Session Automatic
Teste 4	Sessao criada por fenómenos a moni	15/11/2010 at 17:50 to 15/11/2010 at 17:55	Showcase	end	Start Session Stop Session Automatic

Próxima Página

WISE-MUSE PROJECT

WISE-MAN ACTIVITY MONITORING

2010

STOP

Figura IV.19: Listagem das sessões colaborativas.

IV.4.1.4.Exportar Dados

O utilizador pode visualizar o histórico de cada sessão utilizando a interface que se encontra na Figura IV.20. Poderá escolher qual a sessão que deseja gerar o histórico, escolhendo na *combo box* "Choose Session", e também qual a data de início e de fim da recepção das amostras a incluir no histórico. Depois de escolher as opções, é gerado o relatório ao carregar no botão "Generate Session Historic". O relatório será guardado em *C:\Collaborative Sessions Reports\Sessions Report*, sendo o documento *MS Word* guardado com o identificador da sessão, adicionando a data e hora da criação.

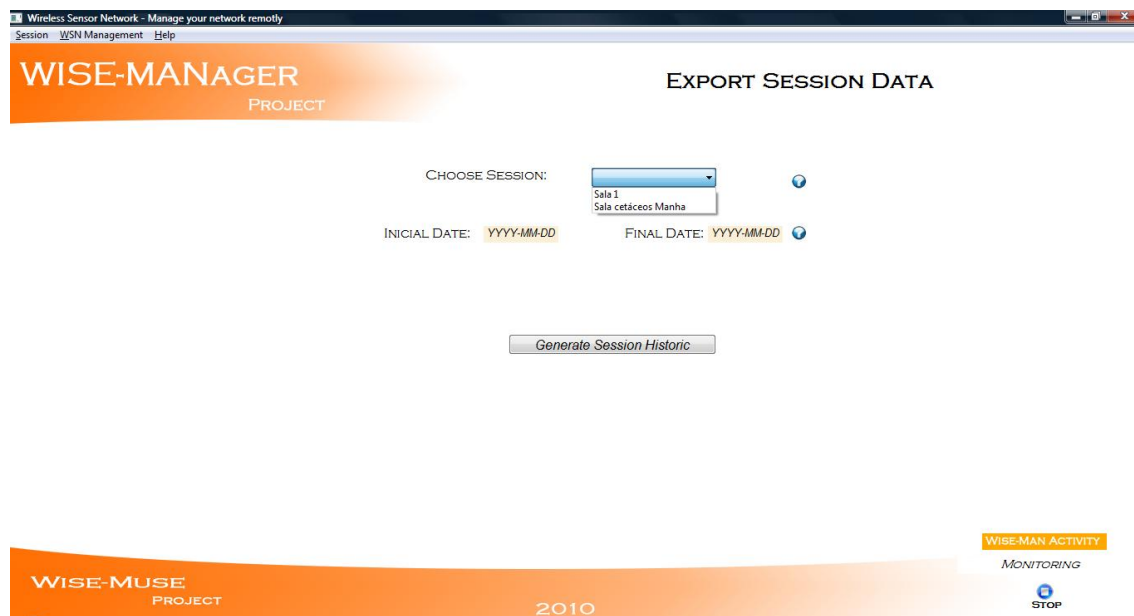


Figura IV.20: Interface da criação de um histórico das sessões.

Um exemplo de um ficheiro gerado encontra-se no anexo B.

IV.4.1.5. Apagar Sessões

O utilizador tem a possibilidade de apagar as sessões da base de dados, como podemos observar na Figura IV.21 em que é apresentada a lista de todas as sessões, com algumas informações acerca das mesmas. Para apagar a sessão, basta clicar em "*Delete Session*", aparecendo duas mensagens de confirmação, sendo que uma delas está na Figura IV.22. Esta figura mostra a quantidade de amostras que a sessão monitorizou, e também serve para evitar que o utilizador cometa erros ao eliminar sessões que não deseja.

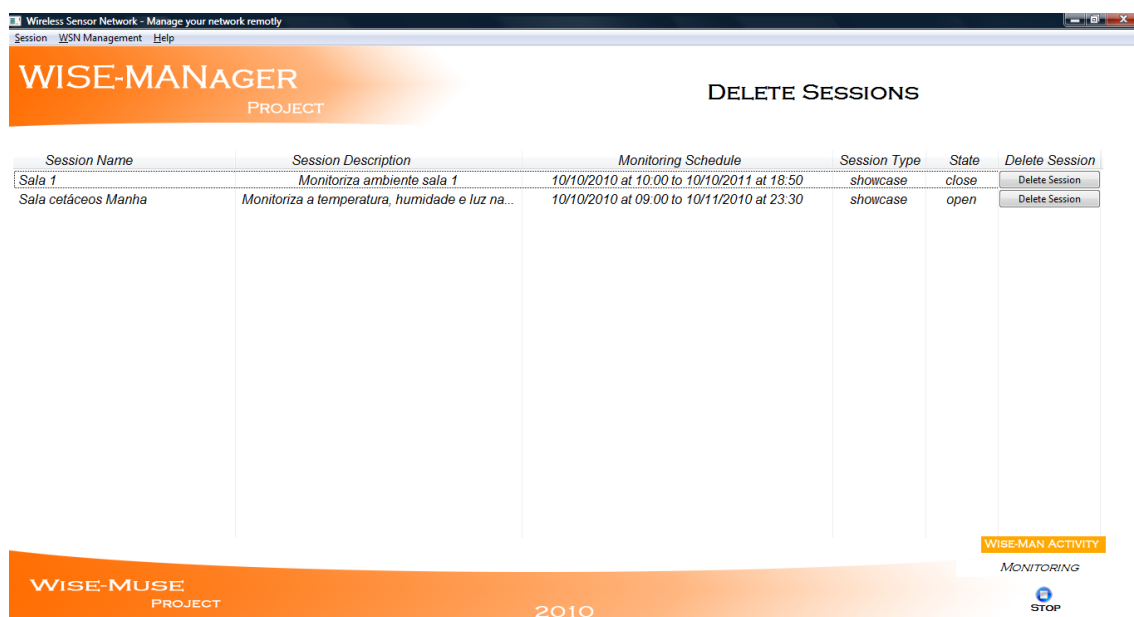


Figura IV.21: Interface de apagar sessões.

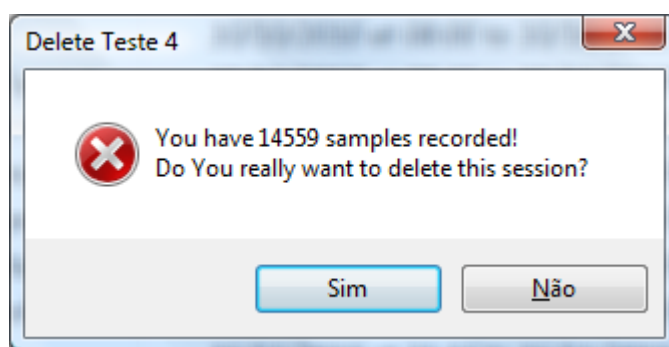


Figura IV.22: Mensagem de informação da quantidade de amostras guardadas.

IV.4.2. WSN Management

No menu *WSN Management*, o utilizador tem a possibilidade de interrogar toda a rede, sabendo informações como o ID da rede, o canal da rede e quais os dispositivos que a compõem.

IV.4.2.1. Conexão com a WSN

Antes de comunicar com a WSN, é necessário criar uma ligação entre a plataforma e a WSN.

No canto inferior direito, o utilizador pode conectar-se à rede XBee, carregando no botão “Start”, de forma a iniciar a monitorização. A Figura IV.12 mostra o local exacto do botão. Ao carregar nesse botão, é apresentada a Figura IV.23:

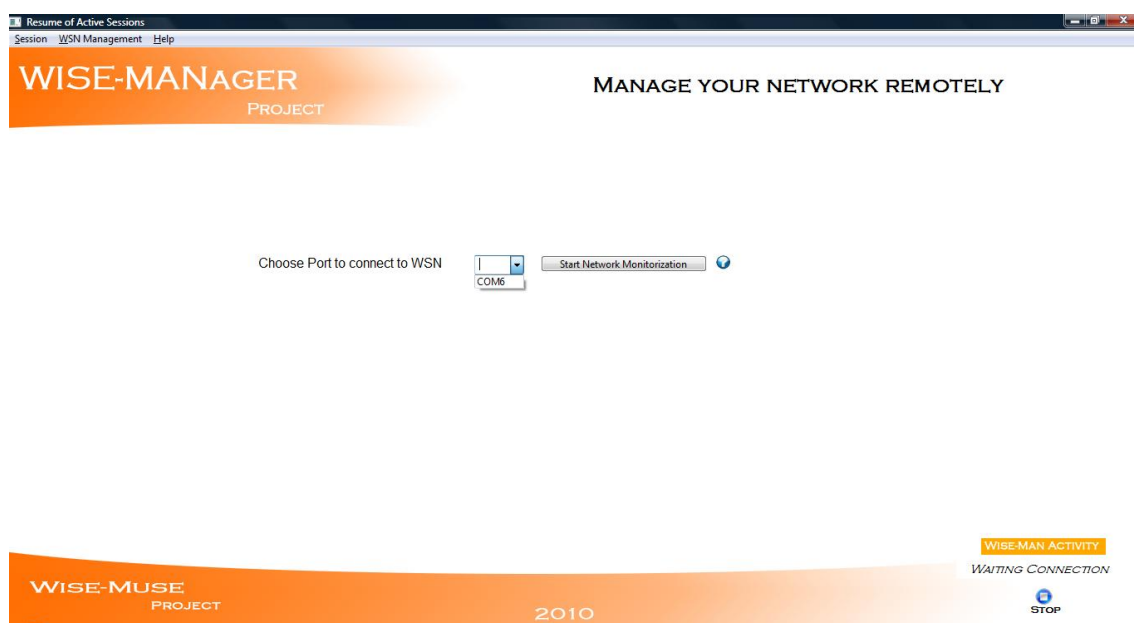


Figura IV.23: Efectuar a conexão com a rede XBee.

Observando a Figura IV.23, verifica-se que apenas é necessário escolher na *combo box* a porta COM em que o coordenador encontra-se, de forma a receber os dados que a rede monitoriza. Esta tarefa apenas é executada uma vez, visto que a conexão mantém-se aberta enquanto o sistema estiver activo.

IV.4.2.2. Comandos Utilizados

Depois de criar a instância da classe *XBee*, na Figura IV.7, podem ser utilizados comandos para a comunicação com a rede. Assim, foram utilizados diversos comandos na gestão da rede:

Na opção *WSN Info* utilizou-se os comandos *OP*, *CH* e *ND*, de forma a descobrir o ID da rede, o canal e a quantidade de dispositivos da rede, respectivamente.

Na opção *WSN Devices*, foram utilizados os comandos *NC*, *NI* e *ND*, para saber quantos filhos cada nó ainda poderia ter, para alterar o nome do nó e obter todas as informações sobre todos os dispositivos da rede, respectivamente. Os comandos *OP*, *CH* e *NI* são enviados de forma síncrona, visto que é recebida uma resposta ao comando enviado. Por outro lado, o comando *ND* é enviado de forma assíncrona, porque este comando é um pedido para que todos os dispositivos da rede enviem as suas informações (quais os seus endereços, o identificador, entre outras). Estes enviam os pacotes que são recebidos por um *listener*, criado anteriormente, e pode ser visualizado na Figura IV.31.

A descrição dos comandos é efectuada de seguida:

OP - Operating PAN

Este comando permite ler o valor da *PAN* (*Personal Area Network*), enviando de seguida esse valor. O *PAN ID* terá que ser único, ou seja, não poderá haver duas redes com o mesmo *ID*. É criado um comando que é enviado para o coordenador da rede, que tem o endereço de 64 bits *0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00*, com o comando *OP*, como podemos visualizar na Figura IV.24. Este comando é criado utilizando a *API XBee* através da classe *ZNetRemoteAtRequest* que é utilizada para a criação de todos os comandos *AT*. Para a criação do comando, é necessário construir o pacote a enviar, dizendo qual a frame a utilizar, neste caso *0x17*, qual o endereço de destino (de 64 bits e 16 bits), e qual o comando a enviar, neste caso o *OP*.

Na Figura IV.24 encontra-se a criação do comando, com a utilização da classe *ZNetRemoteAtRequest*:

```
ZNetRemoteAtRequest requestOP = new ZNetRemoteAtRequest(coordenador, "OP");
```

Figura IV.24: Criação do comando "OP".

O envio do comando é efectuado utilizando a classe *ZNetRemoteAtResponse* com o método *sendSynchronous*, como podemos observar na Figura IV.25, enviando o comando que foi criado na Figura IV.24 e um tempo de *timeout*, ou seja, passado esse tempo irá ser apresentado um erro no envio do comando.

```
ZNetRemoteAtResponse responseOP = (ZNetRemoteAtResponse) xbee.sendSynchronous(requestOP, 10000);
```

Figura IV.25: Envio e recepção do comando "OP".

CH - Operating Channel

Este comando é utilizado para saber qual o canal que a rede utiliza para a comunicação entre os módulos *RF*, sendo que o envio e recepção dos comandos é em tudo semelhante ao comando *OP*. A criação deste comando encontra-se na Figura IV.26, enquanto que o envio, e respectiva resposta estão na Figura IV.27.

```
ZNetRemoteAtRequest requestCH = new ZNetRemoteAtRequest(coordenador, "CH");
```

Figura IV.26: Criação do comando "CH".

```
ZNetRemoteAtResponse responseCH = (ZNetRemoteAtResponse) xbee.sendSynchronous(requestCH, 10000);
```

Figura IV.27: Envio e recepção do comando "CH".

NC - Number of Children

O objectivo deste comando é mostrar quantos *end-devices* ainda podem-se conectar ao coordenador ou aos *routers*, como podemos observar no campo *number of*

remaining children na Figura IV.33. O coordenador da rede pode ser pai de dez *end-devices*, enquanto que cada router pode ter doze *end-devices* conectados a si.

Para a execução deste comando, é necessário saber os endereços dos dispositivos da rede. Por isso, é enviado primeiro o comando *ND*, descrito posteriormente, de forma a obter estes mesmos endereços. O comando *NC* é importante porque permite observar se a rede pode ser facilmente expansível, ou se é necessário colocar mais routers. Na Figura IV.28 está representado como é efectuado o envio do comando *NC*.

```
int i=0;
while(i<networkNodes)
{
    addr64=nodeDiscoverInt.nd[i].getNodeAddress64();

    ZNetRemoteAtRequest request = new ZNetRemoteAtRequest(addr64, "NC");
    ZNetRemoteAtResponse response = (ZNetRemoteAtResponse) xbee.sendSynchronous(request, 10000);

    int[] NCValue = response.getValue();
    nd[i].setNumChild(NCValue[0]);
    i++;
}
```

Figura IV.28: Envio do comando NC para todos os dispositivos da rede.

Observando a Figura IV.28, verificamos que a variável *networkNodes* contém o número de dispositivos da rede, sendo que é enviado para todos o comando *NC*, de forma a saber a quantidade de dispositivos que ainda podem receber. Depois de receber o valor, na variável *response*, este é retirado e guardado na variável *nd*, que é do tipo *NodeDiscover*, o número de filhos de cada dispositivo. A classe *NodeDiscover* guarda todos os nós descobertos na pesquisa efectuada através do comando *ND*.

NI – Node Identifier

O comando *NI* permite ler e escrever o nome do dispositivo, sendo que só podem ser utilizados caracteres *ASCII*. Este comando é utilizado quando o utilizador decide mudar o nome do dispositivo, que podemos observar na Figura IV.33. Para este comando ficar gravado no *XBee* é necessário enviar outros dois comandos, o *AC* – *Apply Changes*, que é para as mudanças serem aplicadas no *XBee* e o *WR* – *Write*, que é para depois de desligar o dispositivo, o mesmo continuar com o novo identificador, ou seja, escreve na memória do dispositivo. Na Figura IV.29 encontra-se o código que permite efectuar a alteração do identificador do dispositivo:

```
int[] nodeIdentInt= new int[nodeIdent[node].getText().length()];
for ( int i = 0; i < nodeIdent[node].getText().length(); ++i ) {
    char c = nodeIdent[node].getText().charAt( i );
    int j = (int) c;
    nodeIdentInt[i]=j;
}

ZNetRemoteAtRequest requestNI = new ZNetRemoteAtRequest(nd[node].getNodeAddress64(), "NI",nodeIdentInt );
xbee.sendSynchronous(requestNI, 10000);

ZNetRemoteAtRequest requestAC = new ZNetRemoteAtRequest(nd[node].getNodeAddress64(), "AC");
xbee.sendSynchronous(requestAC, 10000);

ZNetRemoteAtRequest requestWR = new ZNetRemoteAtRequest(nd[node].getNodeAddress64(), "WR");
xbee.sendSynchronous(requestWR, 10000);
```

Figura IV.29: Envio do comando NI para o dispositivo.

Observando a Figura IV.29, verificamos, no início, que é necessário efectuar a mudança de *String* para *int*, devido ao *XBee* apenas efectuar a leitura de inteiros. Feito isto, são enviados os vários comandos necessários para que esta operação tenha sucesso, o *NI*, o *AC* e o *WR*.

ND - Node Discovery

Este comando permite verificar quais são os dispositivos da rede, assim como vários dos seus parâmetros, como o endereço de 16 bits e 64 bits, o endereço do pai (*parent address*) e o tipo de dispositivo (coordenador, *router* ou *end-device*).

O comando *ND*, que se encontra na Figura IV.30, é enviado de forma assíncrona, ou seja, envia o comando e não recebe imediatamente a resposta. É criado um *listener*, representado na Figura IV.31, que recebe os pacotes que são enviados pelos dispositivos para o coordenador.

```
xbee.sendAsynchronous(new AtCommand("ND"));
```

Figura IV.30: Envio do comando “ND”.

```
xbee.addPacketListener(new PacketListener()
{
    public void processResponse(XBeeResponse response)
    {
        if(response.getApiId() == ApiId.AT_RESPONSE)
        {
            nodeDiscoverInt.receberNodeDiscovery(response);
        }
    }
});
```

Figura IV.31: *Packet Listener* para receber os pacotes da rede.

IV.4.2.3. WSN Info

Ao escolher esta opção serão apresentadas algumas informações da rede, como está apresentado na Figura IV.32. As informações apresentadas são o *ID* da rede, o canal da rede e quais os dispositivos que compõem a rede.

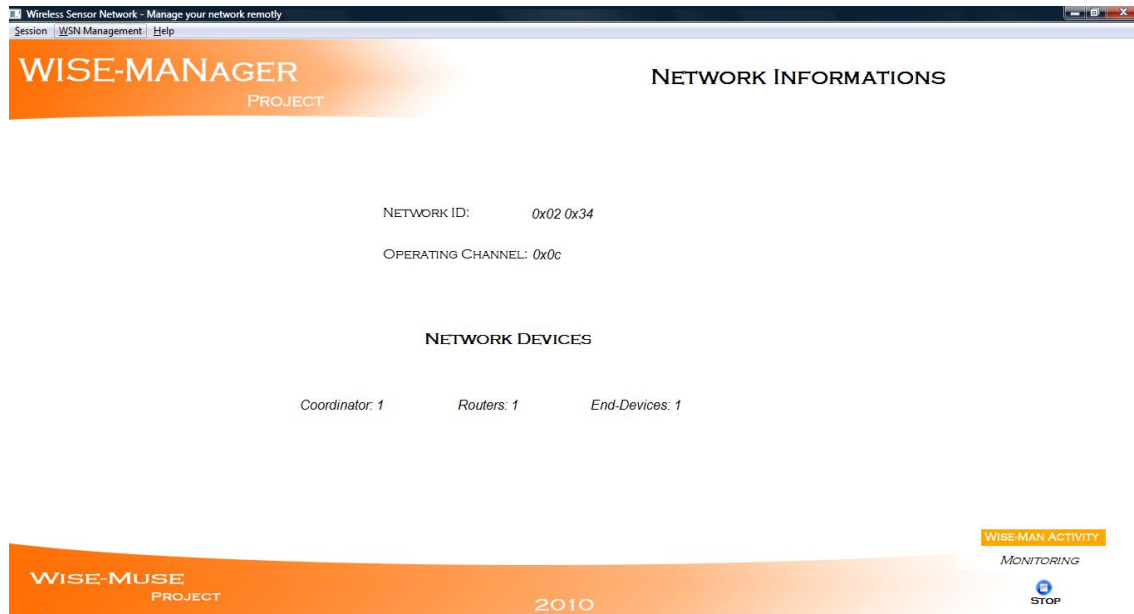


Figura IV.32: Informações da rede.

IV.4.2.4. WSN Device

Na Figura IV.33 estão os dispositivos que se encontram na rede no momento que foi escolhida a opção *WSN Device* no menu *WSN Management*.

O utilizador pode alterar o identificador do dispositivo, alterando para isso o nome na *textbox* do campo *Node Identifier*, e carregando no botão "*Change Node Ident*".

Para obter estas informações dos dispositivos é necessário o envio do comando *ND*, descrito anteriormente, de forma a recolher os parâmetros para preencher a Figura IV.33.

Network Nodes

Session WSN Management Help

WISE-MANAGER
PROJECT

NETWORK DEVICES

Node Identifier	16-bit Address	64-bit Address	Parent Address	Device Type	Number of remaning children	Change Node Identifier
Coord	0x00 0x00	0x00 0x13 0xa2 0x00 0x40 0x3c 0x4c 0x76	0xff 0xfe	COORDINATOR	9	Change Node Ident
No 1	0x01 0x02	0x00 0x13 0xa2 0x00 0x40 0x55 0xa4 0x78	0x00 0x00	END_DEVICE	0	Change Node Ident
Router 1	0x02 0xec	0x00 0x13 0xa2 0x00 0x40 0x2d 0xd7 0xca	0xff 0xfe	ROUTER	12	Change Node Ident

WISE-MAN ACTIVITY

MONITORING

WISE-MUSE
PROJECT

2010

STOP

Figura IV.33: Dispositivos da rede.

IV.5. TRABALHOS RELACIONADOS

Como foi visto no capítulo II, do estado da arte, existem diversas ferramentas de monitorização e gestão da rede, que foram apresentadas nas Tabelas II.3 e II.4. Nas Tabelas II.5 e II.6 encontram-se as comparações ao nível do que as ferramentas permitem personalizar ao nível da monitorizar e a capacidade das mesmas em interrogar a rede, respectivamente. Comparando essas ferramentas com o sistema *WISE-MANager* obteve-se as Tabelas IV.3, IV.4, IV.5 e IV.6.

Tabela IV.3: Comparação dos parâmetros de visualização entre as ferramentas e o sistema *WISE-MANager*.

Visualização Ferramenta	Monitorização em tempo real	Visualização dos nós da rede	Visualização da topologia da rede	Descoberta automática de nós
Tyni DB	Sim	Sim	Sim	Sim
MonSense	Sim	Sim	Sim	Sim
Mote-View	Sim	Sim	Sim	Não
MANNA	Sim	Sim	Sim	Sim
BOSS	Sim	Sim	Sim	Sim
<i>WISE-MANager</i>	Sim	Sim	Não	Sim

Visualizando a Tabela IV.3 verifica-se que o sistema *WISE-MANager* cumpre os parâmetros de visualização das ferramentas de visualização de redes de sensores sem fios.

Tabela IV.4: Comparação dos parâmetros de gestão entre as ferramentas e o sistema *WISE-MANager*.

Gestão Ferramenta	Histórico	Gestão da rede	Gerar Relatórios	Criar Sessões	Consultar Sessões	Monitorizar sessões
Tiny DB	Sim	Não	Não	Não	Não	Não
MonSense	Sim	Sim	Não	Não	Não	Não
Mote-View	Sim	Sim	Sim	Não	Não	Não
MANNA	Não	Sim	Não	Não	Não	Não
BOSS	Não	Sim	Não	Não	Não	Não
<i>WISE-MANager</i>	Sim	Sim	Sim	Sim	Sim	Sim

O sistema *WISE-MANager*, além de permitir ao utilizador gerar relatórios e histórico, efectua a gestão da rede e também permite a criação e gestão de sessões.

Tabela IV.5: Escolher o que monitorizar na WSN.

Monitorização Ferramenta	Escolher os nós sensores	Escolher fenómeno	Escolher o intervalo de	Escolher a área a
---	--------------------------	-------------------	-------------------------	-------------------

			tempo	monitorizar
Tiny DB	Não	Não	Não	Não
MonSense	Não	Não	Não	Não
Mote-View	Não	Não	Não	Não
MANNA	Não	Sim	Não	Sim
BOSS	Não	Não	Não	Não
WISE-MANager	Sim	Sim	Sim	Sim

Uma das grandes vantagens do sistema *WISE-MANager* é a possibilidade do utilizador poder personalizar totalmente a monitorização da WSN.

Tabela IV.6: Interrogar a rede.

Interrogar Ferramenta	Informações da WSN	Dispositivos da WSN	Topologia da Rede	Gestão da Energia
Tiny DB	Não	Sim	Não	Não
MonSense	Não	Não	Não	Não
Mote-View	Não	Não	Não	Não
MANNA	Não	Sim	Sim	Sim
BOSS	Sim	Sim	Sim	Sim
WISE-MANager	Sim	Sim	Não	Sim

Observando a Tabela IV.6, observa-se que o sistema *WISE-MANager* permite recolher todas as informações da WSN, assim como efectuar a gestão de energia dos nós sensores.

Portanto, depois desta análise, podemos afirmar que o sistema *WISE-MANager* tem as seguintes vantagens:

- Criação de sessões colaborativas;
- Consulta de sessões colaborativas;
- Geração de históricos;
- Monitorização de sessões;
- Portabilidade.
- Personalização dos nós a monitorizar;
- Personalização dos fenómenos a monitorizar;
- Personalização do intervalo de tempo a monitorizar.

Como desvantagens, podemos verificar que este sistema não permite visualizar a topologia da rede, apenas mostrando uma lista dos nós da rede, sendo que é um dos trabalhos futuros na extensão deste sistema.

O sistema *WSN-MANager* permite ao utilizador efectuar uma manipulação directa sobre a *WSN*, visto que pode iniciar e parar a monitorização da rede, além de interrogar a rede para obter dados específicos. Além disto, o utilizador pode visualizar algumas informações da rede, assim como quais os dispositivos que compõem a rede, podendo alterar o identificador destes dispositivos. Ele também tem possibilidade de gerir as sessões colaborativas, criando-as para recolher amostras da rede, podendo depois visualizar essas amostras.

Assim, podemos concluir que o sistema *WISE-MANager* propõe um novo conceito de colaboração, onde o utilizador pode modelar e interagir com a rede de uma forma mais estreita e através de uma ferramenta gráfica em tempo real. Este tipo de colaboração pode trazer vantagens como:

- Monitorização personalizada, o utilizador pode definir livremente que parâmetros as sessões monitorizam;
- O utilizador pode definir quais nós sensores actuarão nas sessões colaborativas, aumentando a flexibilidade e o grau de controlo e coordenação do utilizador sobre a rede;
- O utilizador ao controlar a colaboração também pode influenciar a topologia da rede;
- Definir diferentes tipos de sessões colaborativas, paralela, sequencial, síncrona ou assíncrona;
- Aumentar a eficiência no caso de *WSN* heterogéneas activando os nós com os parâmetros de interesse, ou seja, activando determinadas sessões.
- Poupança de energia por parte dos sensores, activando-se só quando é necessário.

No futuro pretendemos estender para redes homogéneas, controlando o sensor directamente através do sistema *WISE-MANager*.

IV.6. CONCLUSÃO

A fase da implementação foi dividida em duas partes. Na primeira, foi criada uma aplicação, *RSSI Level*, que permite a visualização do nível de sinal de um dispositivo *XBee*. Esta aplicação foi utilizada no estudo de propagação do sinal realizado no Museu da Baleia. A segunda parte foi o desenvolvimento do sistema *WISE-MANager*, que é utilizado nas sessões colaborativas e na gestão da rede.

O estudo de propagação do sinal no museu da baleia foi efectuado através de oito testes, permitindo assim obter o melhor *deployment* da rede de sensores sem fios a instalar naquele local. Esta rede será composta por um total de 73 nós, que monitorizam temperatura, humidade, luz, monóxido e dióxido de carbono, sendo também possível saber o estado actual dos bloqueadores das portas.

O sistema *WISE-MANager* permite a recepção de todos os pacotes enviados pela *WSN*, incluindo os pacotes enviados pelos bloqueadores e os enviados pelos nós sensores que monitorizam o ambiente. Além disto, permite a criação, manipulação e visualização de sessões colaborativas, ou seja, permite ao utilizador criar sessões por fenómeno a monitorizar, nós da rede ou intervalo temporal, de forma a receber e guardar as diferentes amostras da rede. Além disto, esta ferramenta permite efectuar a gestão da rede, através do envio e recepção de comandos síncronos e assíncronos, visualizando quais os dispositivos que fazem parte da mesma e também observar algumas informações da rede.

No próximo capítulo serão apresentados os diversos testes, e respectivos resultados, efectuados ao sistema *WISE-MANager*.

V. TESTES E RESULTADOS

V.1. INTRODUÇÃO

Neste capítulo são apresentados os testes efectuados ao sistema *WISE-MANager* desenvolvido neste projecto de Mestrado. O objectivo destes testes é verificar que a plataforma cumpre os requisitos pré-definidos, ou seja, que o utilizador consegue fazer todas as tarefas utilizando este sistema. Verificou-se também o comportamento do sistema na recepção dos dados provenientes da *WSN*, a sua interpretação e armazenamento na base de dados, assim como observar que o sistema permite a visualização, em tempo real, da recepção dessas amostras das diferentes sessões. Também foram efectuados testes de gestão de rede, para que seja possível observar quais os componentes que se encontram na rede.

V.2. AMBIENTE DOS TESTES

Efectuou-se os testes no Laboratório de Informática 3 da Universidade da Madeira, testando o sistema *WISE-MANager* perante uma *WSN* operacional que monitoriza temperatura, humidade e luminosidade.

Nos testes, foram utilizados um computador portátil com o *software* necessário instalado, ou seja, o sistema *WISE-MANager* instalado, assim como os drivers necessários para a conexão do dispositivo *XBee* que actua como coordenador da *WSN*, que podemos observar na Figura V.1. Do lado da *WSN*, foram utilizados cinco dispositivos, três routers e dois nós sensores *WISE-MUSE*, que podemos observar nas Figuras V.2 e V.3 respectivamente. Estes dispositivos foram construídos previamente no âmbito do projecto *WISE-MUSE* por Bruno Gouveia. Os nós sensores estão configurados para enviar as amostras dos fenómenos de dez em dez segundos, enquanto que o router envia amostras assincronamente.



Figura V.1: Coordenador da rede.



Figura V.2: Router.



Figura V.3: Nó sensor.

V.3. TESTES REALIZADOS

Foram efectuados diversos testes utilizando o *WISE-MANager*, no ambiente descrito anteriormente. Nos testes, foram utilizados diversos componentes:

Hardware utilizado:

1. Computador pessoal;
2. Módulo XBee configurado como coordenador da WSN;
3. Dois nós sensores *WISE-MUSE*;
4. Três nós routers *WISE-MUSE*.

Software utilizado no computador pessoal:

1. Servidor Apache 2.2.14;
2. MySQL 5.1.41;
3. PhpMyAdmin 3.2.4;
4. Sistema *WISE-MANager* 1.0.

Os pontos 1, 2 e 3 do *software* utilizado, foram instalados utilizando o pacote *Apache Friends XAMPP* 1.7.3.

Para os testes, é necessário que todo o *software* esteja a executar. Criou-se a ligação entre a plataforma e a WSN utilizando a porta COM13, utilizando previamente um driver que permita essa conexão.

V.3.1. Teste 1

Neste teste utilizou-se um computador portátil, o coordenador da rede e um nó sensor, como podemos observar na Figura V.4:

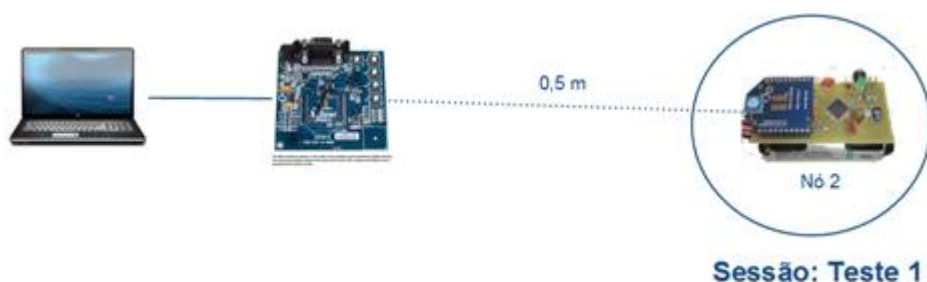


Figura V.4: Esquema de montagem da rede no Teste 1.

O objectivo deste teste foi verificar se o sistema *WISE-MANager* tinha capacidade de, automaticamente, iniciar e terminar uma sessão colaborativa dentro de um intervalo de tempo definido. Para isto, criou-se uma sessão, chamada de teste 1, que inicia a monitorização às 9:15 e termina às 9:20. O nó sensor 2 irá efectuar a

monitorização dos fenómenos temperatura, humidade e luminosidade para esta sessão. Desta forma, criou-se a sessão, que se encontra na Figura V.5. Os resultados dos fenómenos monitorizados podem ser acompanhados no menu *View Active Sessions*, na opção *Monitoring Data*. Na Figura V.6 encontram-se os resultados finais da monitorização.

WISE-MANAGER PROJECT

CREATE A COLLABORATIVE SESSION

Name of session:

Description of session:

Schedule of Monitoration

Start Date	End Date	Start Hour	End Hour
<input type="text" value="15/11/2010"/>	<input type="text" value="15/11/2010"/>	<input type="text" value="09:15"/>	<input type="text" value="09:20"/>

Nodes choosen for session

Name	Type
No 2	Te;Hu;Li

Session Type

☒ **Showcase**
 Temperature
 Humidity
 Light

☐ **Room**
 Temperature
 Humidity
 Light
 Carbon Monoxide
 Carbon Dioxide

☐ **Personalized**
☐ Temperature
☐ Humidity
☐ Light
☐ Carbon Monoxide
☐ Carbon Dioxide
☐ Emergency Doors

Figura V.5: Criação da sessão colaborativa utilizada no teste 1.

Monitoring Results for session: Teste 1

Date	Node	Temperature	Humidity	Light	Battery
2010-11-15 09:20:09.0	2	22.76 °	53.2458 %	557.251	3.16787
2010-11-15 09:19:59.0	2	22.76 °	53.2458 %	516.968	3.17109
2010-11-15 09:19:49.0	2	22.74 °	53.1167 %	248.413	3.16787
2010-11-15 09:19:39.0	2	22.76 °	53.149 %	550.537	3.15176
2010-11-15 09:19:29.0	2	22.75 °	53.149 %	308.838	3.16787
2010-11-15 09:19:19.0	2	22.78 °	53.2135 %	255.127	3.17432
2010-11-15 09:19:09.0	2	22.78 °	53.0845 %	255.127	3.17109
2010-11-15 09:18:59.0	2	22.78 °	53.0199 %	416.26	3.17109
2010-11-15 09:18:49.0	2	22.76 °	53.0199 %	248.413	3.17432
2010-11-15 09:18:39.0	2	22.78 °	53.0199 %	241.699	3.17432
2010-11-15 09:18:29.0	2	22.76 °	53.0522 %	248.413	3.17109
2010-11-15 09:18:19.0	2	22.76 °	53.0845 %	275.269	3.17432
2010-11-15 09:18:09.0	2	22.76 °	53.1813 %	375.977	3.17432
2010-11-15 09:17:59.0	2	22.74 °	53.1813 %	543.823	3.17754
2010-11-15 09:17:49.0	2	22.75 °	53.149 %	557.251	3.17432
2010-11-15 09:17:39.0	2	22.74 °	53.149 %	456.543	3.17109
2010-11-15 09:17:29.0	2	22.73 °	53.149 %	255.127	3.17109
2010-11-15 09:17:19.0	2	22.73 °	53.149 %	248.413	3.17432
2010-11-15 09:17:09.0	2	22.73 °	53.1813 %	288.696	3.17754
2010-11-15 09:16:59.0	2	22.74 °	53.149 %	550.537	3.17754
2010-11-15 09:16:49.0	2	22.74 °	53.149 %	543.823	3.17754
2010-11-15 09:16:39.0	2	22.72 °	53.0845 %	557.251	3.17432
2010-11-15 09:16:29.0	2	22.72 °	53.1167 %	302.124	3.17754
2010-11-15 09:16:19.0	2	22.74 °	53.0199 %	563.965	3.17432

Figura V.6: Resultados obtidos com a sessão Teste 1.

Observando os resultados da Figura V.6 verifica-se que a sessão monitorizou de forma correcta entre as 9:15 e as 9:20, monitorizando também os fenómenos pretendidos, neste caso temperatura, humidade e luminosidade. O valor da bateria serve como parâmetro informativo, que é útil para saber quando é necessário trocar a mesma.

V.3.2. Teste 2

Neste teste utilizou-se os mesmos componentes que no teste anterior, apenas adicionando mais um nó sensor, como podemos observar pela montagem da Figura V.7:

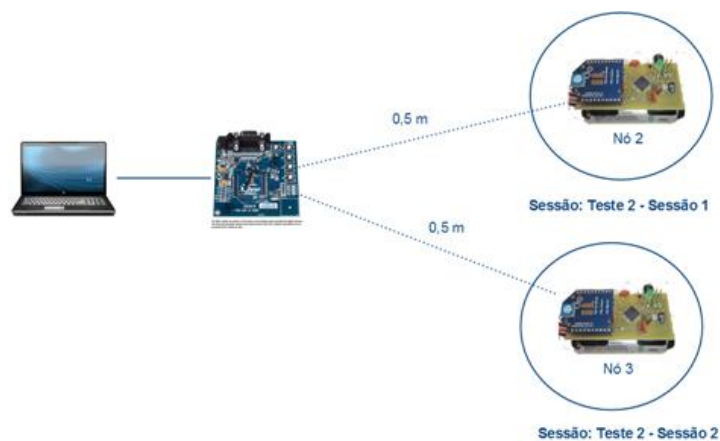
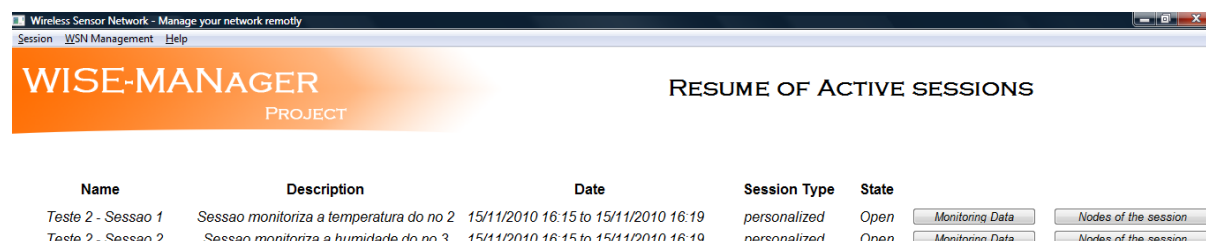


Figura V.7: Esquema de montagem da rede no Teste 2.

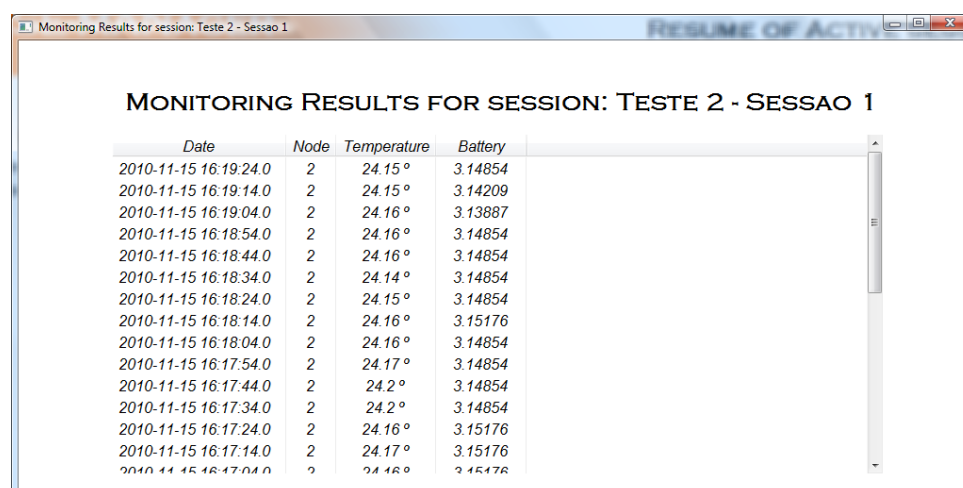
Neste teste iremos observar se as sessões apenas monitorizam o que foi pedido, assim como apenas recebem os pacotes que a elas são destinados, utilizando as duas sessões, Teste 2 - Sessão 1 e Teste 2 - Sessão 2, que monitorizam em paralelo. Uma irá conter o nó 2 e monitoriza a temperatura, e outra irá conter o nó 3 e irá monitorizar a humidade. A criação das sessões é efectuada da mesma forma que o teste anterior. Na Figura V.8 podemos observar que as duas sessões encontram-se activas, e verificamos que irão ambas monitorizar no mesmo horário, apesar de monitorizarem fenómenos diferentes.



Name	Description	Date	Session Type	State		
Teste 2 - Sessao 1	Sessao monitoriza a temperatura do no 2	15/11/2010 16:15 to 15/11/2010 16:19	personalized	Open	Monitoring Data	Nodes of the session
Teste 2 - Sessao 2	Sessao monitoriza a humidade do no 3	15/11/2010 16:15 to 15/11/2010 16:19	personalized	Open	Monitoring Data	Nodes of the session

Figura V.8: Sessões activas no Teste 2.

Os resultados das sessões Teste 2 - Sessao 1 e Teste 2 - Sessao 2, encontram-se nas Figuras V.9 e V.10 respectivamente:



Date	Node	Temperature	Battery
2010-11-15 16:19:24.0	2	24.15 °	3.14854
2010-11-15 16:19:14.0	2	24.15 °	3.14209
2010-11-15 16:19:04.0	2	24.16 °	3.13887
2010-11-15 16:18:54.0	2	24.16 °	3.14854
2010-11-15 16:18:44.0	2	24.16 °	3.14854
2010-11-15 16:18:34.0	2	24.14 °	3.14854
2010-11-15 16:18:24.0	2	24.15 °	3.14854
2010-11-15 16:18:14.0	2	24.16 °	3.15176
2010-11-15 16:18:04.0	2	24.16 °	3.14854
2010-11-15 16:17:54.0	2	24.17 °	3.14854
2010-11-15 16:17:44.0	2	24.2 °	3.14854
2010-11-15 16:17:34.0	2	24.2 °	3.14854
2010-11-15 16:17:24.0	2	24.16 °	3.15176
2010-11-15 16:17:14.0	2	24.17 °	3.15176
2010-11-15 16:17:04.0	2	24.16 °	3.15176

Figura V.9: Resultados da monitorização da Sessão: Teste 2 - Sessao 1.

Monitoring Results for session: Teste 2 - Sessao 2

Date	Node	Humidity	Battery
2010-11-15 16:19:17.0	3	57.2635 %	2.45244
2010-11-15 16:19:07.0	3	57.1689 %	2.45566
2010-11-15 16:18:57.0	3	57.1689 %	2.45244
2010-11-15 16:18:47.0	3	57.2635 %	2.45244
2010-11-15 16:18:37.0	3	57.4211 %	2.45566
2010-11-15 16:18:27.0	3	57.3896 %	2.45566
2010-11-15 16:18:17.0	3	57.3896 %	2.45244
2010-11-15 16:18:07.0	3	57.3581 %	2.45566
2010-11-15 16:17:57.0	3	57.2951 %	2.45566
2010-11-15 16:17:47.0	3	57.3581 %	2.45244
2010-11-15 16:17:37.0	3	57.3266 %	2.45244
2010-11-15 16:17:27.0	3	57.232 %	2.45566
2010-11-15 16:17:17.0	3	57.232 %	2.45566
2010-11-15 16:17:07.0	3	57.2635 %	2.45566
2010-11-15 16:16:57.0	3	57.2635 %	2.45566

Figura V.10: Resultados da monitorização da Sessão: Teste 2 – Sessao 2.

Observando as figuras V.9 e V.10 verifica-se que obtivemos os resultados esperados, visto que na Sessão 1 apenas temos os valores da temperatura utilizando o nó 2, e na Sessão 2 obtivemos os valores da humidade utilizando o nó 3.

V.3.3. Teste 3

Neste teste iremos utilizar os dois nós sensores, na mesma sessão, juntamente com o coordenador da rede, da forma descrita na Figura V.11:

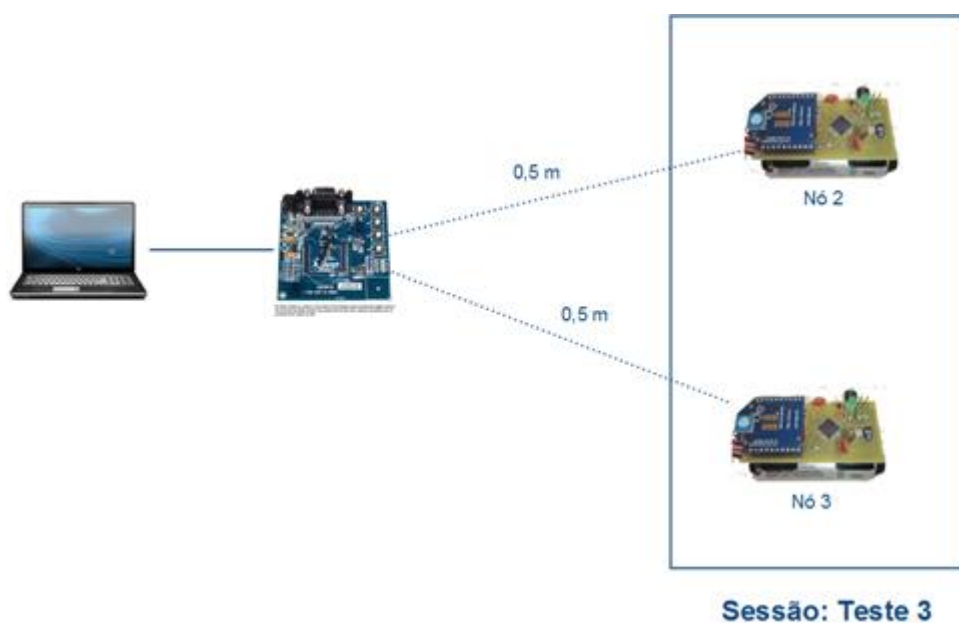


Figura V.11: Esquema de montagem da rede no Teste 3.

O objectivo deste teste é verificar se a sessão consegue diferenciar os pacotes de diferentes nós e colocá-los na mesma sessão. Os resultados encontram-se na Figura V.12:

Monitoring Results for session: Teste 3

Date	Node	Temperature	Humidity	Light	Battery
2010-11-15 16:36:19.0	2	24.34 °	54.7548 %	349.121	3.14209
2010-11-15 16:36:16.0	3	24.33 °	56.2199 %	335.693	2.45244
2010-11-15 16:36:09.0	2	24.34 °	54.7548 %	315.552	3.14531
2010-11-15 16:36:06.0	3	24.31 °	56.2199 %	342.407	2.45244
2010-11-15 16:35:59.0	2	24.32 °	54.7868 %	483.398	3.11953
2010-11-15 16:35:56.0	3	24.31 °	56.2199 %	335.693	2.45244
2010-11-15 16:35:49.0	2	24.33 °	54.9466 %	255.127	3.14531
2010-11-15 16:35:46.0	3	24.32 °	56.2833 %	167.847	2.45244
2010-11-15 16:35:39.0	2	24.32 °	55.0743 %	234.985	3.14531
2010-11-15 16:35:36.0	3	24.34 °	56.4417 %	328.979	2.44922
2010-11-15 16:35:29.0	2	24.32 °	55.0743 %	483.398	3.15176
2010-11-15 16:35:26.0	3	24.3 °	56.5368 %	302.124	2.45244
2010-11-15 16:35:19.0	2	24.32 °	55.0743 %	496.826	3.12598
2010-11-15 16:35:16.0	3	24.31 °	56.5684 %	154.419	2.45244
2010-11-15 16:35:09.0	2	24.3 °	55.0743 %	443.445	3.13249

Figura V.12: Resultados da monitorização utilizando a sessão Teste 3.

Observando a Figura V.12 verifica-se que a sessão Teste 3 conseguiu receber as amostras dos dois nós sensores, durante o tempo de monitorização da mesma.

V.3.4. Teste 4

A sessão utilizada neste teste foi criada utilizando como critério os fenómenos a monitorizar. Neste caso, os dois nós monitorizam a temperatura, humidade e luz. Assim, e de forma a testar o router, o esquema da rede foi o descrito na Figura V.13:

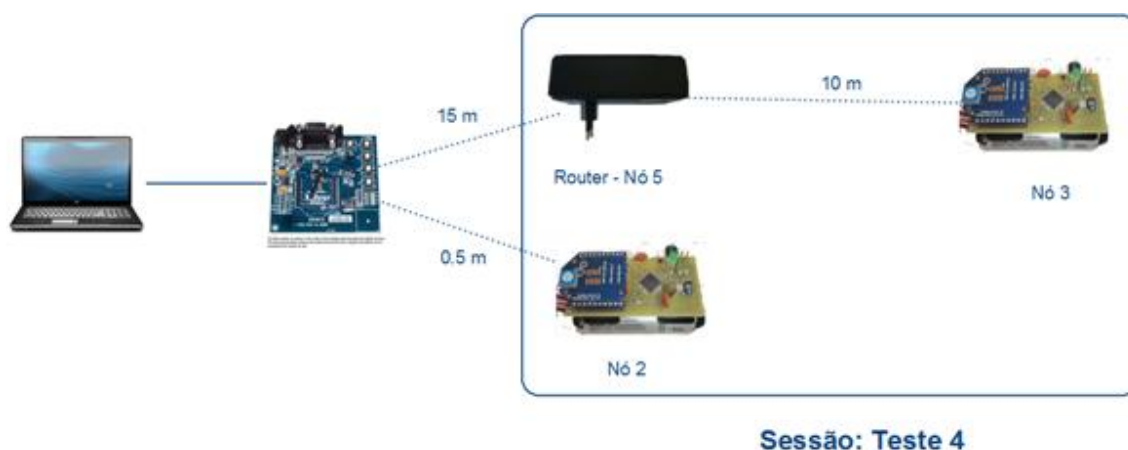
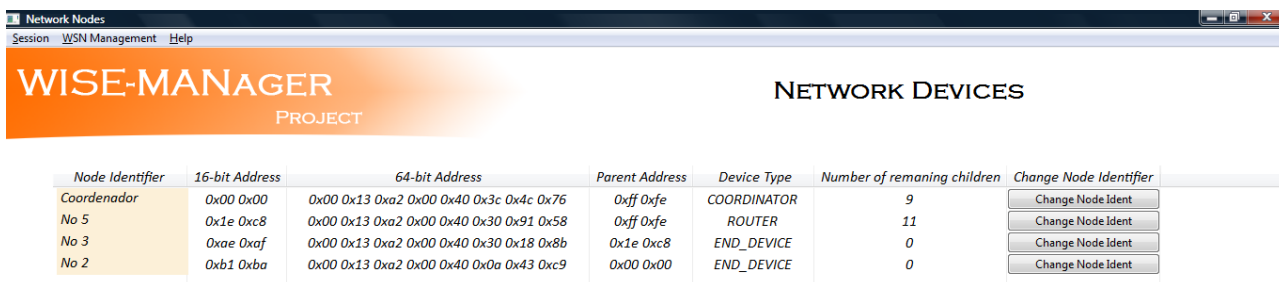


Figura V.13: Esquema da rede utilizado no Teste 4.

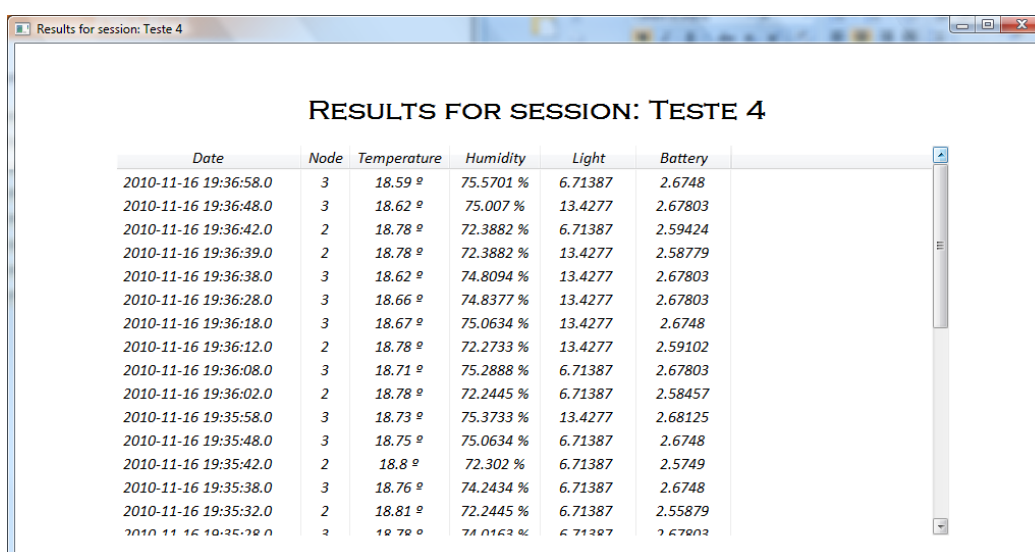
Desta forma testamos se o router efectua o encaminhamento de pacotes de forma correcta, ou seja, se as amostras do nó 3 chegam ao coordenador. Na Figura V.14 encontra-se a descrição dos dispositivos da rede:



Node Identifier	16-bit Address	64-bit Address	Parent Address	Device Type	Number of remaining children	Change Node Identifier
Coordenador	0x00 0x00	0x00 0x13 0xa2 0x00 0x40 0x3c 0x4c 0x76	0xff 0xfe	COORDINATOR	9	<button>Change Node Ident</button>
No 5	0x1e 0xc8	0x00 0x13 0xa2 0x00 0x40 0x30 0x91 0x58	0xff 0xfe	ROUTER	11	<button>Change Node Ident</button>
No 3	0xae 0xaf	0x00 0x13 0xa2 0x00 0x40 0x30 0x18 0x8b	0x1e 0xc8	END_DEVICE	0	<button>Change Node Ident</button>
No 2	0xb1 0xba	0x00 0x13 0xa2 0x00 0x40 0x0a 0x43 0xc9	0x00 0x00	END_DEVICE	0	<button>Change Node Ident</button>

Figura V.14: Dispositivos da rede no Teste 4.

Na Figura V.15 encontram-se os resultados desta sessão:



Date	Node	Temperature	Humidity	Light	Battery
2010-11-16 19:36:58.0	3	18.59 °	75.5701 %	6.71387	2.6748
2010-11-16 19:36:48.0	3	18.62 °	75.007 %	13.4277	2.67803
2010-11-16 19:36:42.0	2	18.78 °	72.3882 %	6.71387	2.59424
2010-11-16 19:36:39.0	2	18.78 °	72.3882 %	13.4277	2.58779
2010-11-16 19:36:38.0	3	18.62 °	74.8094 %	13.4277	2.67803
2010-11-16 19:36:28.0	3	18.66 °	74.8377 %	13.4277	2.67803
2010-11-16 19:36:18.0	3	18.67 °	75.0634 %	13.4277	2.6748
2010-11-16 19:36:12.0	2	18.78 °	72.2733 %	13.4277	2.59102
2010-11-16 19:36:08.0	3	18.71 °	75.2888 %	6.71387	2.67803
2010-11-16 19:36:02.0	2	18.78 °	72.2445 %	6.71387	2.58457
2010-11-16 19:35:58.0	3	18.73 °	75.3733 %	13.4277	2.68125
2010-11-16 19:35:48.0	3	18.75 °	75.0634 %	6.71387	2.6748
2010-11-16 19:35:42.0	2	18.8 °	72.302 %	6.71387	2.5749
2010-11-16 19:35:38.0	3	18.76 °	74.2434 %	6.71387	2.6748
2010-11-16 19:35:32.0	2	18.81 °	72.2445 %	6.71387	2.55879
2010-11-16 19:35:28.0	3	18.78 °	74.0163 %	6.71387	2.67803

Figura V.15: Resultados da sessão Teste 4.

Ao observar a Figura V.15, verifica-se que o encaminhamento de pacotes por parte do nó 5 encontra-se a funcionar de forma perfeita, visto que a sessão recebe amostras de todos os nós.

V.3.5. Teste 5

Utilizando a configuração descrita na Figura V.13, iremos testar se a sessão apenas monitoriza quando se encontra no estado activo, para isto iremos parar a monitorização da sessão, ou seja, colocar a opção *stop* no *actual session state* da sessão. A sessão criada irá monitorizar das 10:25 às 10:30, sendo que o botão de *stop* será carregado as 10:27 e a sessão ficará em *Stop* até as 10:28, ou seja, não irá monitorizar durante este minuto. De seguida, será posta outra vez no estado automático, ou seja, irá monitorizar até as 10:30. Os resultados desta sessão encontram-se na Figura V.16.

Results for session: Teste 5

RESULTS FOR SESSION: TESTE 5

Date	Node	Temperature	Humidity	Light	Battery
2010-11-16 10:28:59.0	2	23.12 °	54.7229 %	261.841	2.58779
2010-11-16 10:28:56.0	3	21.08 °	61.1307 %	53.7109	2.44277
2010-11-16 10:28:49.0	2	23.13 °	54.6589 %	476.685	2.58779
2010-11-16 10:28:46.0	3	21.09 °	61.4081 %	67.1387	2.43955
2010-11-16 10:28:39.0	2	23.12 °	54.6909 %	530.396	2.58135
2010-11-16 10:28:36.0	3	21.09 °	61.5927 %	67.1387	2.44277
2010-11-16 10:28:29.0	2	23.12 °	54.6589 %	248.413	2.56846
2010-11-16 10:28:26.0	3	21.08 °	61.4697 %	60.4248	2.43955
2010-11-16 10:28:19.0	2	23.1 °	54.6909 %	402.832	2.59102
2010-11-16 10:28:16.0	3	21.1 °	61.5004 %	67.1387	2.43955
2010-11-16 10:26:59.0	2	23.08 °	54.8188 %	355.835	2.59102
2010-11-16 10:26:56.0	3	21.14 °	61.5927 %	53.7109	2.44277
2010-11-16 10:26:49.0	2	23.09 °	54.8507 %	523.682	2.60391
2010-11-16 10:26:46.0	3	21.14 °	61.5312 %	67.1387	2.43955
2010-11-16 10:26:39.0	2	23.08 °	54.8507 %	537.109	2.58779
2010-11-16 10:26:36.0	3	21.14 °	61.5004 %	67.1387	2.44277

Figura V.16: Resultados da sessão Teste 5.

Ao observar a Figura V.16, verifica-se que no minuto 10:27 não ocorreu a monitorização, ou seja, o objectivo deste teste foi alcançado.

Com esta funcionalidade, é permitido ao utilizador parar instantaneamente a monitorização de uma sessão, assim como iniciar uma sessão a qualquer momento.

V.3.6. Teste 6

Neste teste, iremos utilizar a disposição da rede utilizada anteriormente, mas com uma configuração diferente, que pode ser observada na Figura V.17. Irão ser utilizadas duas sessões, que estão organizadas em sequência, ou seja, uma irá começar quando a outra acaba. Ambas as sessões são do tipo *Showcase*, ou seja, monitorizam os fenómenos ambientais de temperatura, humidade e luz.

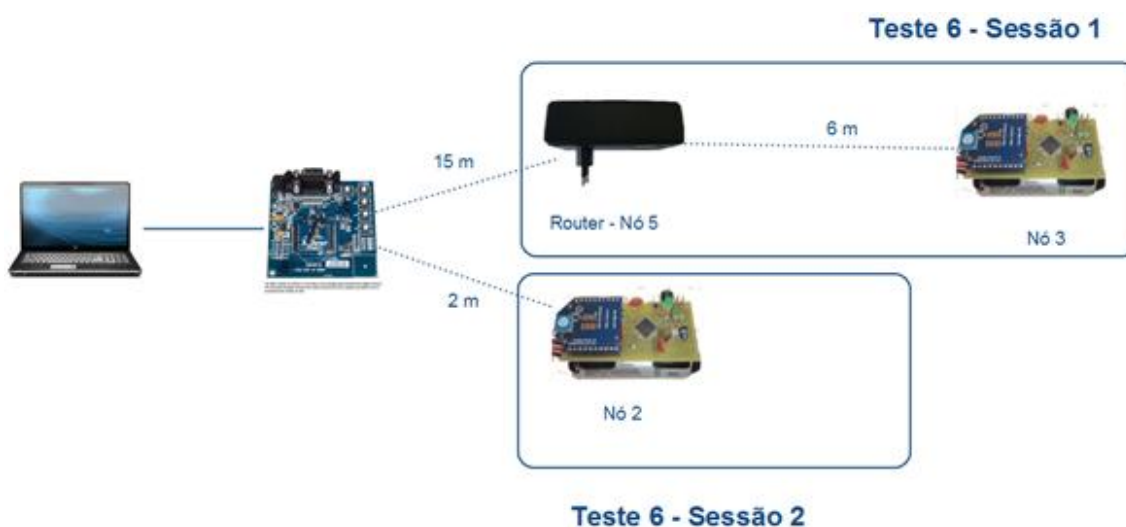


Figura V.17: Esquema da rede utilizado no Teste 6.

Testes e Resultados

O objectivo deste teste é verificar se as duas sessões eram capazes de monitorizar em sequência, ou seja, uma a seguir a outra. Nas figuras V.18 e V.19 encontram-se as sessões “Teste 6 – Sessão 1” e “Teste 6 – Sessão 2”, na altura em que cada uma delas encontrava-se a monitorizar.



Name	Description	Date	Session Type	State
Teste 6 - Sessao 1	Sessão 1/2 em sequencia	16/11/2010 21:07 to 16/11/2010 21:11	showcase	Open

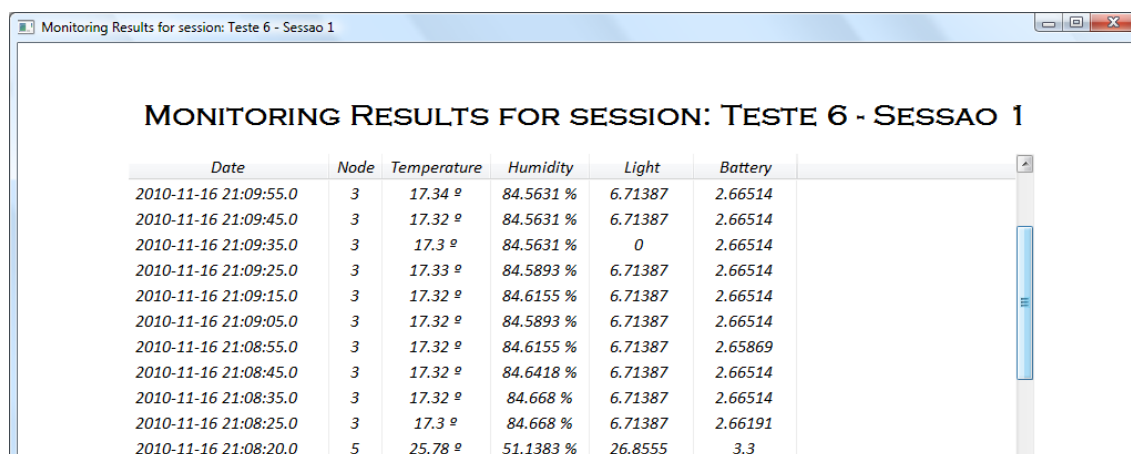
Figura V.18: Sessão: Teste 6 – Sessão 1.



Name	Description	Date	Session Type	State
Teste 6 - Sessao 2	Sessão 2/2 em sequencia	16/11/2010 21:11 to 16/11/2010 21:14	showcase	Open

Figura V.19: Sessão: Teste 6 – Sessão 2.

Observando os resultados das Figuras V.20 e V.21 verificamos que a sessão “Teste 6 – Sessao 2” monitoriza depois da sessão “Teste 6 – Sessão 1”, e que não existe monitorização em simultâneo das ambas, verificando que o teste foi um sucesso.



Date	Node	Temperature	Humidity	Light	Battery
2010-11-16 21:09:55.0	3	17.34 °	84.5631 %	6.71387	2.66514
2010-11-16 21:09:45.0	3	17.32 °	84.5631 %	6.71387	2.66514
2010-11-16 21:09:35.0	3	17.3 °	84.5631 %	0	2.66514
2010-11-16 21:09:25.0	3	17.33 °	84.5893 %	6.71387	2.66514
2010-11-16 21:09:15.0	3	17.32 °	84.6155 %	6.71387	2.66514
2010-11-16 21:09:05.0	3	17.32 °	84.5893 %	6.71387	2.66514
2010-11-16 21:08:55.0	3	17.32 °	84.6155 %	6.71387	2.65869
2010-11-16 21:08:45.0	3	17.32 °	84.6418 %	6.71387	2.66514
2010-11-16 21:08:35.0	3	17.32 °	84.668 %	6.71387	2.66514
2010-11-16 21:08:25.0	3	17.3 °	84.668 %	6.71387	2.66191
2010-11-16 21:08:20.0	5	25.78 °	51.1383 %	26.8555	3.3

Figura V.20: Resultados da sessão Teste 6 – Sessao 1.

Results for session: Teste 6 - Sessão 2

Date	Node	Temperature	Humidity	Light	Battery
2010-11-16 21:13:44.0	2	19 °	74.4417 %	6.71387	2.57168
2010-11-16 21:13:24.0	2	19.02 °	74.4417 %	6.71387	2.57168
2010-11-16 21:13:04.0	2	19.02 °	74.3568 %	6.71387	2.56846
2010-11-16 21:12:24.0	2	19.02 °	74.4701 %	6.71387	2.55557
2010-11-16 21:12:04.0	2	19.04 °	74.4417 %	6.71387	2.58135

Figura V.21: Resultados da sessão Teste 6 – Sessao 2.

V.3.7. Teste 7

Neste teste o nó 2 encontra-se em duas sessões distintas executando-se em paralelo. A sessão 1 irá monitorizar a temperatura enquanto que a sessão 2 monitoriza a humidade. O esquema da rede está representado na Figura V.22.

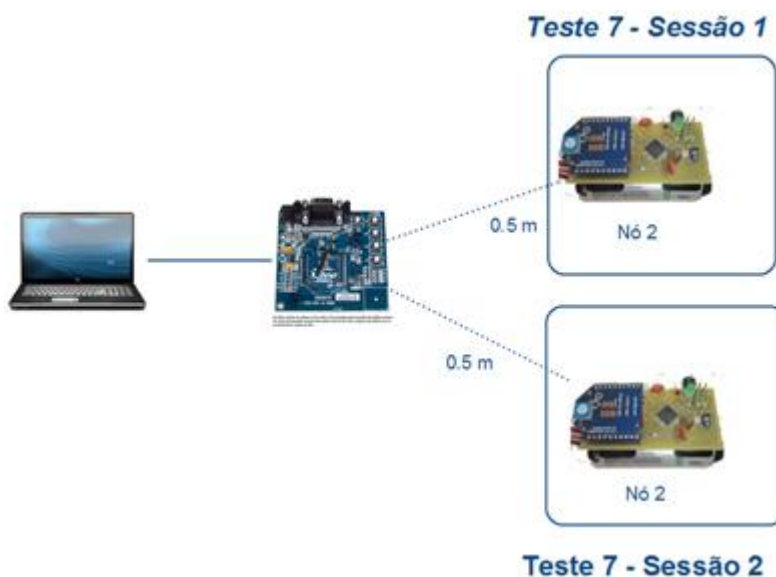


Figura V.22: Esquema da rede utilizado no Teste 7.

Os resultados deste teste encontram-se nas Figuras V.23 e V.24:

Results for session: Teste 1

RESULTS FOR SESSION: TESTE 1

Date	Node	Temperature	Battery
2010-11-22 09:49:20.0	2	25.24 °	2.6168
2010-11-22 09:48:53.0	2	25.24 °	2.58457
2010-11-22 09:48:23.0	2	25.27 °	2.62002
2010-11-22 09:48:13.0	2	25.27 °	2.58457
2010-11-22 09:47:53.0	2	25.23 °	2.62324
2010-11-22 09:47:43.0	2	25.22 °	2.61035
2010-11-22 09:47:33.0	2	25.22 °	2.62324
2010-11-22 09:47:23.0	2	25.23 °	2.62002
2010-11-22 09:47:13.0	2	25.22 °	2.62002
2010-11-22 09:46:53.0	2	25.23 °	2.62002
2010-11-22 09:46:43.0	2	25.23 °	2.62002
2010-11-22 09:46:33.0	2	25.2 °	2.62324
2010-11-22 09:46:23.0	2	25.17 °	2.62324
2010-11-22 09:46:13.0	2	25.16 °	2.62324
2010-11-22 09:46:03.0	2	25.14 °	2.62324
2010-11-22 09:45:53.0	2	25.12 °	2.62324

Figura V.23: Resultado da sessão Teste 7 – Sessão 1.

Results for session: Teste 2

RESULTS FOR SESSION: TESTE 2

Date	Node	Humidity	Battery
2010-11-22 09:49:20.0	2	67.0083 %	2.6168
2010-11-22 09:48:53.0	2	67.0976 %	2.58457
2010-11-22 09:48:23.0	2	67.2759 %	2.62002
2010-11-22 09:48:13.0	2	67.7504 %	2.58457
2010-11-22 09:47:53.0	2	68.1349 %	2.62324
2010-11-22 09:47:43.0	2	67.365 %	2.61035
2010-11-22 09:47:33.0	2	67.0678 %	2.62324
2010-11-22 09:47:23.0	2	67.0381 %	2.62002
2010-11-22 09:47:13.0	2	67.0381 %	2.62002
2010-11-22 09:46:53.0	2	67.2462 %	2.62002
2010-11-22 09:46:43.0	2	67.3353 %	2.62002
2010-11-22 09:46:33.0	2	67.3056 %	2.62324
2010-11-22 09:46:23.0	2	67.3353 %	2.62324
2010-11-22 09:46:13.0	2	67.5133 %	2.62324
2010-11-22 09:46:03.0	2	67.543 %	2.62324
2010-11-22 09:45:53.0	2	67.6319 %	2.62324

Figura V.24: Resultado da sessão Teste 7 – Sessão 2.

Ao observar estas figuras, verifica-se que as sessões monitorizam em paralelo.

V.3.8. Teste 8

Para saber quais são os dispositivos da rede, é necessário escolher o menu *WSN Device*, enquanto que ao escolher o menu *WSN Info* são apresentadas as

informações da rede. Será isto que o Teste 8 irá testar, ou seja, irá verificar se a informação é consistente, além de permitir ao utilizador alterar o nome do dispositivo. O esquema de rede utilizado está descrito na Figura V.25:

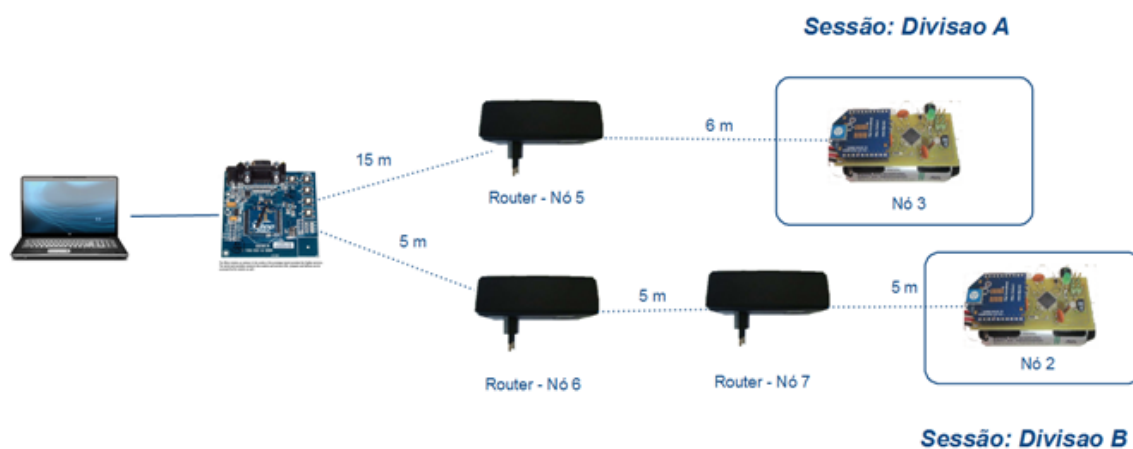


Figura V.25: Esquema de montagem da rede.

Na Figura V.26 encontra-se as informações da rede e na Figura V.27 encontra-se a listagem dos dispositivos da rede.

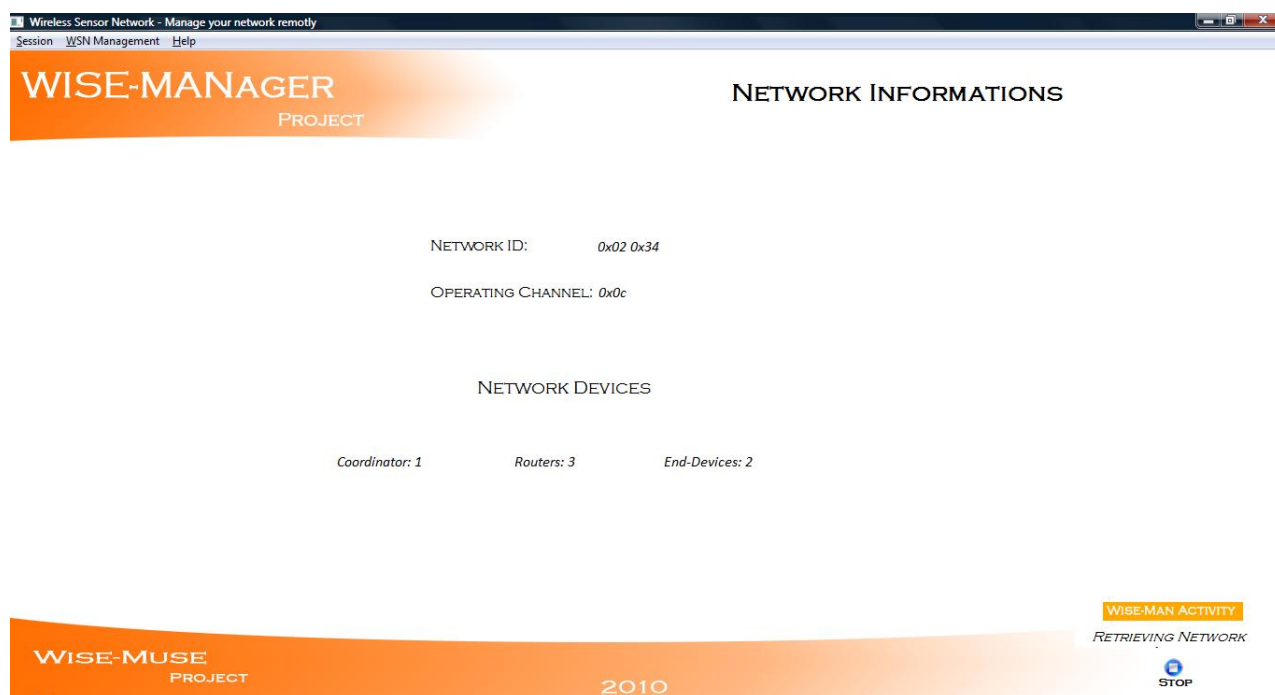


Figura V.26: Informações da rede de sensores sem fios.

The screenshot shows the 'WISE-MANAGER NETWORK DEVICES' window. It contains a table with the following columns: Node Identifier, 16-bit Address, 64-bit Address, Parent Address, Device Type, Number of remaning children, and Change Node Identifier. The table lists several nodes, including a Coordinator and several numbered nodes (No 5, No 2, No 6, No 7, No 3) with their respective addresses and device types (ROUTER, END_DEVICE). Each row has a 'Change Node Identifier' button next to it.

Node Identifier	16-bit Address	64-bit Address	Parent Address	Device Type	Number of remaning children	Change Node Identifier
Coordenador	0x00 0x00	0x00 0x13 0xa2 0x00 0x40 0x3c 0x4c 0x76	0xff 0xfe	COORDINATOR	10	Change Node Identifier
No 5	0x1e 0xc8	0x00 0x13 0xa2 0x00 0x40 0x30 0x91 0x58	0xff 0xfe	ROUTER	11	Change Node Identifier
No 2	0x40 0x5f	0x00 0x13 0xa2 0x00 0x40 0x0a 0x43 0xc9	0x70 0xba	END_DEVICE	0	Change Node Identifier
No 6	0xc4 0x7f	0x00 0x13 0xa2 0x00 0x40 0x2d 0xad 0xca	0xff 0xfe	ROUTER	12	Change Node Identifier
No 7	0x70 0xba	0x00 0x13 0xa2 0x00 0x40 0x55 0xa4 0x78	0xff 0xfe	ROUTER	11	Change Node Identifier
No 3	0xbd 0xe9	0x00 0x13 0xa2 0x00 0x40 0x30 0x18 0x8b	0x1e 0xc8	END_DEVICE	0	Change Node Identifier

Figura V.27: Dispositivos da rede de sensores sem fios.

Nas Figuras V.26 e V.27 podemos observar os dispositivos que encontram-se na rede, assim como as características e parâmetros desses dispositivos.

V.3.9.Teste 9

Este será um teste de longa duração, de forma a verificar se o sistema processa de forma correcta as amostras enviadas pela rede durante um largo período de tempo. O esquema de rede utilizado encontra-se na Figura V.23.

Neste teste irão ser monitorizadas duas divisões distintas, uma que é necessário dois routers para ter cobertura, e outra que apenas é necessário um router. A sessão *Divisão A* utiliza o nó 3 e irá monitorizar os fenómenos ambientes de temperatura, humidade e luz. A outra sessão, *Divisão B*, irá monitorizar os mesmos fenómenos, mas utilizando o nó 2.

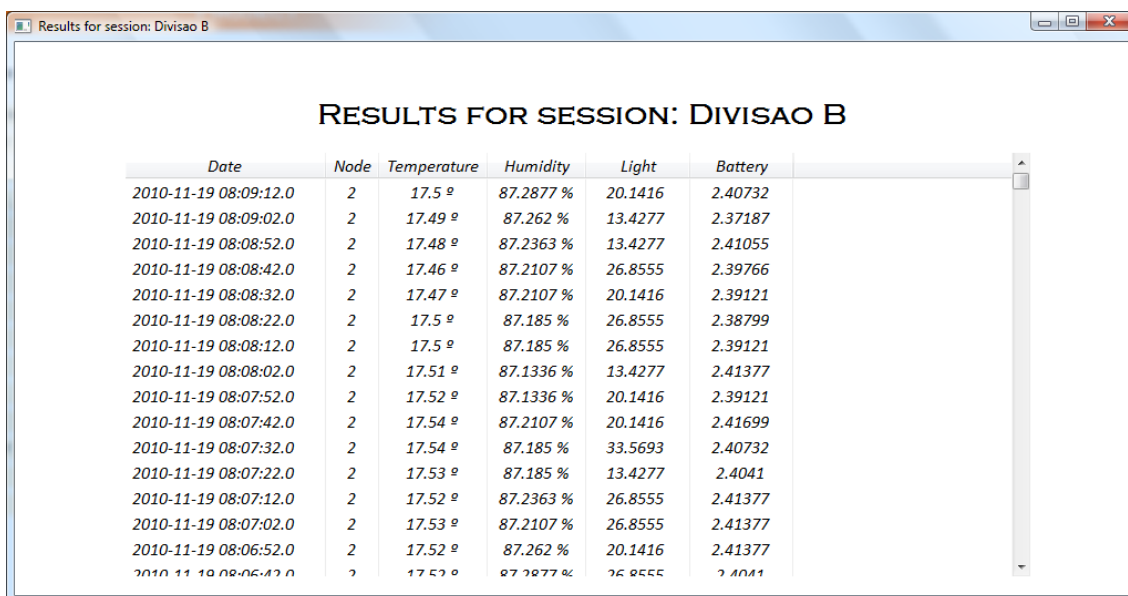
Nas Figuras V.26 e V.26 podemos observar os resultados da sessão *Divisão A* e da sessão *Divisão B*, respectivamente:



RESULTS FOR SESSION: DIVISAO A

Date	Node	Temperature	Humidity	Light	Battery
2010-11-19 08:06:13.0	3	16.5 °	92.8938 %	13.4277	2.64902
2010-11-19 08:06:03.0	3	16.5 °	92.8449 %	13.4277	2.64258
2010-11-19 08:05:43.0	3	16.48 °	92.7961 %	13.4277	2.6458
2010-11-19 08:05:33.0	3	16.5 °	92.7717 %	13.4277	2.64258
2010-11-19 08:05:23.0	3	16.49 °	92.7961 %	13.4277	2.64258
2010-11-19 08:05:13.0	3	16.5 °	92.7961 %	6.71387	2.6458
2010-11-19 08:05:03.0	3	16.48 °	92.8205 %	6.71387	2.6458
2010-11-19 08:04:53.0	3	16.51 °	92.8693 %	13.4277	2.6458
2010-11-19 08:04:43.0	3	16.48 °	92.9425 %	6.71387	2.6458
2010-11-19 08:04:33.0	3	16.49 °	92.9425 %	6.71387	2.63936
2010-11-19 08:04:23.0	3	16.49 °	92.9913 %	6.71387	2.6458
2010-11-19 08:03:23.0	3	16.42 °	92.4046 %	6.71387	2.6458
2010-11-19 08:02:55.0	3	16.42 °	92.3556 %	6.71387	2.6458
2010-11-19 08:02:53.0	3	16.42 °	92.3556 %	6.71387	2.6458
2010-11-19 08:02:03.0	3	16.42 °	92.1838 %	6.71387	2.6458
2010-11-19 08:01:53.0	3	16.42 °	92.1507 %	6.71387	2.6458

Figura V.28: Resultados da sessão Divisão A.



RESULTS FOR SESSION: DIVISAO B

Date	Node	Temperature	Humidity	Light	Battery
2010-11-19 08:09:12.0	2	17.5 °	87.2877 %	20.1416	2.40732
2010-11-19 08:09:02.0	2	17.49 °	87.262 %	13.4277	2.37187
2010-11-19 08:08:52.0	2	17.48 °	87.2363 %	13.4277	2.41055
2010-11-19 08:08:42.0	2	17.46 °	87.2107 %	26.8555	2.39766
2010-11-19 08:08:32.0	2	17.47 °	87.2107 %	20.1416	2.39121
2010-11-19 08:08:22.0	2	17.5 °	87.185 %	26.8555	2.38799
2010-11-19 08:08:12.0	2	17.5 °	87.185 %	26.8555	2.39121
2010-11-19 08:08:02.0	2	17.51 °	87.1336 %	13.4277	2.41377
2010-11-19 08:07:52.0	2	17.52 °	87.1336 %	20.1416	2.39121
2010-11-19 08:07:42.0	2	17.54 °	87.2107 %	20.1416	2.41699
2010-11-19 08:07:32.0	2	17.54 °	87.185 %	33.5693	2.40732
2010-11-19 08:07:22.0	2	17.53 °	87.185 %	13.4277	2.4041
2010-11-19 08:07:12.0	2	17.52 °	87.2363 %	26.8555	2.41377
2010-11-19 08:07:02.0	2	17.53 °	87.2107 %	26.8555	2.41377
2010-11-19 08:06:52.0	2	17.52 °	87.262 %	20.1416	2.41377
2010-11-19 08:06:42.0	2	17.52 °	87.2877 %	26.8555	2.4041

Figura V.29: Resultados da sessão Divisão B.

Neste teste, que ocorreu entre as 23:00 e as 8:10, foram recebidas cerca de 2800 amostras em cada sessão.

Neste teste as amostras foram enviadas de dez em dez segundos, fazendo com que a quantidade de amostras enviadas seja muito alta. Quando a WSN estiver implementada, serão enviadas amostras a cada dez minutos. Assim sendo, este teste representou o envio de amostras de sensivelmente 22 dias da WSN a ser implementada no museu.

O sistema *WISE-MANager* conseguiu receber e tratar de todas as mensagens enviadas pelos dispositivos da rede, sendo que o único problema poderá ser alguma

demora na apresentação dos resultados (que se encontram nas Figuras V.28 e V.29) devido à grande quantidade de amostras das sessões.

V.3.10. Teste 10

Neste teste irá ser utilizada uma rede composta por um coordenador e um *end-device*, que será o bloqueador. O diagrama da rede encontra-se na Figura V.30:



Figura V.30: Esquema da rede no teste 10.

Foi criada a sessão Teste 10, que monitoriza das 18:00 as 18:10 o estado deste bloqueador. Os bloqueadores podem-se encontrar em três estados diferentes: *open*, *close* ou *emergency*. Sempre que o estado do bloqueador se altere, é enviada uma mensagem para a rede com essa alteração. Na Figura V.31 encontra-se o registo dessas alterações:

MONITORING RESULTS FOR SESSION: TESTE 10			
Date	Node	Bloq	Battery
2010-11-23 18:09:40.0	6	open	3.3
2010-11-23 18:09:30.0	6	close	3.3
2010-11-23 18:09:20.0	6	open	3.3
2010-11-23 18:07:02.0	6	close	3.3
2010-11-23 18:07:00.0	6	emergency	3.3
2010-11-23 18:06:00.0	6	close	3.3
2010-11-23 18:05:00.0	6	open	3.3

Figura V.31: Resultados obtidos no teste 10.

V.4. CONCLUSÃO

Após terminados os testes, verificou-se que a plataforma *WISE-MANager* foi capaz de receber as amostras da rede, efectuar o seu tratamento e guardá-las, em tempo real. As sessões utilizam essas amostras recolhidas da rede, mostrando-as ao utilizador em tempo real.

A recepção dos pacotes da rede foi efectuada de forma correcta, visto que não foi perdido nenhum, sendo depois colocados na base de dados *WISE-MUSE* para serem utilizados por esta ferramenta, e por outras englobadas no projecto *WISE-MUSE*. As amostras de temperatura e humidade encontram-se dentro dos parâmetros esperados, enquanto que as amostra de luminosidade foram afectadas pela passagem de pessoas no laboratório, aquando dos testes.

Verificou-se que a *thread* utilizada para receber dados e enviar comandos para a rede em simultâneo funciona de forma correcta, permitindo ao utilizador receber amostras dos fenómenos e efectuar a gestão da rede de sensores sem fios.

Conclui-se que a ferramenta apresenta uma boa robustez, visto que foi capaz de receber e tratar de mais de 5000 mensagens, que foram enviadas por 2 nós sensores, a cada dez segundos, durante nove horas.

Com os diversos testes cumpridos com sucesso, verificou-se que a ferramenta cumpre todos os requisitos propostos inicialmente.

VI. CONCLUSÃO

VI.1. INTRODUÇÃO

Neste ultimo capítulo, são apresentadas as principais contribuições deste projecto de Mestrado na área de gestão de sessões colaborativas em redes de sensores sem fios. Será também descrito os trabalhos futuros a serem efectuados nesta área, a partir da utilização deste projecto. Finalmente, é feito um balanço final deste projecto de Mestrado.

VI.2. PRINCIPAIS CONTRIBUIÇÕES

Este projecto de mestrado, que se encontra englobado no projecto *WISE-MUSE* teve como contribuições principais:

- Estudo de propagação do sinal realizado no Museu da Baleia, que permitiu obter um correcto *deployment* da *WSN* a instalar dentro do museu.
- O desenvolvimento de uma ferramenta que fornece uma interface que permite criar e gerir sessões colaborativas, possibilitando ao utilizador fazer uso de um novo tipo de colaboração com a *WSN*, ou seja, ter uma colaboração mais estreita com a rede, manipulando e controlando a rede podendo personalizar os parâmetros ambientais a monitorizar, assim como quais os nós sensores e quando estes monitorizam.
- O desenvolvimento de um gestor de *WSN*, que permite visualizar os dispositivos da rede, e alterar alguns parâmetros dos mesmos. Com a sua utilização, pode ser visualizado o estado de todos os dispositivos da rede assim como os seus parâmetros, permitindo desta forma detectar qualquer problema.
- A criação de uma base de dados específica para as sessões colaborativas, permitindo ao utilizador a alteração de todos os parâmetros das sessões colaborativas, assim como, a fácil visualização das amostras recolhidas da *WSN*.
- Elaboração de relatórios internos para o projecto *WISE-MUSE*.
- Elaboração e submissão de um artigo científico na conferência *CNRS 2011 – Portuguese Conference on Wireless Sensor Network*.

VI.3. TRABALHOS FUTUROS

Este projecto poderá ter os seguintes trabalhos futuros:

- Visualização da topologia ou “mapa da rede” interrogando os seus componentes de forma a obter as relações entre eles;
- Gráficos dos dados recolhidos da *WSN*;
- Estender as funcionalidades do sistema *WISE-MANager* para permitir a gestão de *WSN* baseados em outras tecnologias *wireless*;
- Implementar mais comandos para a manipulação da *WSN*;
- Mostrar graficamente o estado das sessões colaborativas.

VI.4. BALANÇO FINAL

No término deste projecto, verifica-se que os principais objectivos propostos foram cumpridos. Foi possível abranger os conhecimentos em diversas áreas, principalmente redes e programação. Foi necessário compreender o funcionamento de diversas tecnologias, e entender de que forma ocorriam as comunicações nessas tecnologias. Isto levou a que os meus conhecimentos nesta área saíssem reforçados.

Depois de compreender o funcionamento das *WSN*, verifica-se que o sistema desenvolvido neste projecto de mestrado está apto a ser utilizado numa *WSN* operacional. Foi muito aliciante desenvolver este sistema, capaz de enviar e receber remotamente comandos da rede de sensores sem fios.

Este projecto foi importante para validar o modelo *CWSN*, utilizando o mesmo para a criação e gestão das sessões em tempo real, sendo que este sistema será utilizado futuramente no Museu da Baleia.

VII. REFERÊNCIAS

- [1] Raghavendra, C. S. et al., "Wireless sensor networks", Springer, 2004.
- [2] Zhao, Feng, et al., "Wireless sensor networks: an information processing approach The Morgan Kaufmann series in networking", Morgan Kaufmann, 2004.
- [3] Brito, L., Rodríguez, "A collaborative Model for Wireless Sensor Networks Applied to Museums' Environmental Monitoring", Mathematics and Engineering (DME), University of Madeira (UMa), Madeira, Portugal.
- [4] Cordeiro, Carlos, et al., "AD HOC & SENSOR NETWORKS Theory and Applications", World Scientific, 2006.
- [5] ZigBee Alliance, "ZigBee: The Choice for Energy Management and Efficiency", June 2007.
- [6] Sohraby, Kazem, et al., "Wireless sensor networks: technology, protocols, and applications", Wiley-Interscience, 2007.
- [7] Girão, Pedro, et al., "Wireless Sensor Networks: State of the Art and Future Trends", Instituto de Telecomunicações/Instituto Superior Técnico, 2007.
- [8] Malafaya, Hugo, et al., "Sensorização sem fios sobre ZigBee e IEEE 802.15.4", Departamento de Engenharia Electrotécnica e de computadores, Faculdade de Engenharia da Universidade do Porto.
- [9] Craig, William, "ZigBee: "Wireless Control That Simply Works", Program Manager Wireless Communications ZMD America.
- [10] Loureiro, Antonio, et al., "Redes de Sensores Sem Fio", Departamento de Ciência da Computação, Universidade Federal de Minas Gerais.
- [11] Kinney, Patrick, "ZigBee Technology: Wireless Control that Simply Works", Outubro 2003.

- [12] Digi International, "XBee/XBee-Pro ZB RF Modules", 2009.
- [13] Brito, L., Rodríguez, "Hierarchical Modeling of Collaboration in Wireless Sensor Networks", Centro de Ciências Matemáticas (CCM), University of Madeira (UMa), Madeira, Portugal.
- [14] Winnie, Lee, e tal., "Network Management in Wireless Sensor Networks", School of Computer Science & Software Engineering, The University of Western Australia", Australia.
- [15] Pinto, José, et al., "MonSense-Application for Deployment, Monitoring and Control of Wireless Sensor Networks", Faculdade de Engenharia da Universidade do Porto, Portugal.
- [16] S. Farahani, "ZigBee Wireless Networks and Transceivers", Newnes, 2008.
- [17] A. Koubaa, et al., "A time Division Beacon Scheduling Mechanism for IEEE 802.15.4/ZigBee Cluster-Tree Wireless Sensor Networks", Polytechnic Institute of Porto, Portugal, 2007.
- [18] S. Madden, et al. "TinyDB: An Acquisitional Query Processing System for Sensor Networks", Massachusetts Institute of Technology, United States of America, 2005.
- [19] R. Malan, et al. "Defining Non-Functional Requirements", Bredemeyer Consulting, 2001.
- [20] R. Malan et al. "Functional Requirements and Use Cases", Bredemeyer Consulting, 2001.
- [21] M. Dumas, et al. "UML Activity Diagrams as a Workflow Specification Language", Cooperative Information Systems Research Centre Queensland University of Technology, 2001.
- [22] R. Prather, et al. "Decomposition of flowchart schemata", Department of Mathematics and Computer Science, University of Denver, Denver, Colorado 80208, USA, 1981.
- [23] C. Heuser, "Projecto de Banco de Dados", Institute de Informática UFRGS, Sagra Luzzatto, 4ª Edição, 1998.
- [24] G. Booch et al. "Unified Modeling Language User Guide, The", Addison Wesley, October 20, 1998.
- [25] L. Constantine, "Canonical Abstract Prototypes for Abstract Visual and Interaction Design, University of Technology, Sidney, Australia, 2003".
- [26] M. Saleiro, et al., "ZigBee - Uma abordagem prática", Universidade do Algarve, 2006.
- [27] Eclipse, <http://www.eclipse.org>, consultado em 14 de Novembro de 2010.

- [28] PC Magazine Encyclopedia, http://www.pcmag.com/encyclopedia_term/0,2542,t=application+programming+interface&i=37856,00.asp, consultado em 14 de Novembro de 2010.
- [29] SWT, <http://www.eclipse.org/swt/> consultado em 14 de Novembro de 2010.
- [30] L.B. Ruiz, J.M. Nogueira, and A.A.F. Loureiro, "MANNA: A Management Architecture for Wireless Sensor Networks," IEEE Communications Magazine, 2003.
- [31] [H. Song, D. Kim, K. Lee, and J. Sung, "Upnp-Based Sensor Network Management Architecture," 2005.
- [32] Museu da Baleia da Madeira, <http://www.museudabaleia.org>, consultado em 14 de Novembro de 2010.

VIII. ANEXOS

VIII.1. ANEXO A

Foram efectuados diversos testes no Museu da Baleia que são descritos neste anexo.

Para realizar os testes utilizamos dois dispositivos. Um fica configurado como coordenador que envia o comando DB para o outro dispositivo (que pode ser configurado como *router* ou *end-device*), e este envia o valor de *RSSI*. Só utilizamos dois dispositivos porque a introdução de mais dispositivos levava a que o valor de *RSSI* não fosse o real.

Os testes estão documentados de seguida:

VIII.1.1.Primeiro Teste

O primeiro teste foi realizado no dia 22 de Abril de 2010.

O local do coordenador já tinha sido determinado pelos gestores do Museu, pois já tinham um espaço reservado para o mesmo. No local onde o coordenador é colocado ainda não se encontrava revestido de pladur. Para testar a propagação do sinal através deste material realizou-se um teste no museu, num quarto revestido de pladur. Colocou-se o coordenador dentro do quarto e no lado de fora a uma distância de 2m do coordenador ficou o end-device, alimentado a bateria. Verificou-se que o sinal é muito pouco afectado por este tipo de material.

Mediu-se no piso 0 a sala de cetáceos e retirou-se o valor do *RSSI* de dois em dois metros e aproximadamente 2m de altura. Mediu-se também um valor na diagonal a 30 metros do coordenador, de forma a verificar a propagação do sinal no centro da sala. O coordenador foi colocado na canto inferior esquerdo da sala. Os resultados desta medições encontram-se na Figura VIII.1.

VIII.1.2.Segundo teste

O segundo teste foi realizado no dia 10 de Maio de 2010. Neste teste concluímos o estudo de propagação do sinal na sala de cetáceos no piso 0. Verificamos que para o conseguirmos obter sinal na sala de projecção de vídeo necessitávamos de colocar um router entre o coordenador e esta sala. Desta forma, foi criado o router R1. Os valores de *RSSI* obtidos encontram-se na Figura VIII.1.

VIII.1.3.Terceiro teste

O terceiro teste foi realizado a 26 de Maio de 2010, e neste teste mediu-se no piso 0 a sala de projecção de vídeo, sala caça da baleia e átrio. Considerou-se o

coordenador no lugar de R1. A sala de projecção de vídeo mediu-se globalmente de modo a ter uma ideia de como o sinal é propagado. Na sala caça da baleia os gestores do museu pretendem sensores em todas as vitrinas e nas obras de arte. Desta forma, considerou-se estes pontos como “chaves” para determinar o valor do RSSI. No átrio medimos os pontos onde encontrava-se obras de arte. Os resultados encontram-se na Figura VIII.1.

VIII.1.4.Quarto teste

O quarto teste foi realizado a 9 de Junho de 2010 e analisou-se o melhor local para colocar o *router* que encaminha os dados do piso 0 para o piso 1. Neste caso, considerou-se o coordenador na sua sala, no piso 0, e colocamos o dispositivo de teste *end-device*, em vários lugares para analisar a melhor posição. O *router*, R3, ficou localizado na rampa do piso 1. Posteriormente, mediu-se globalmente, no piso 1, o auditório e a sala de exposições temporárias. Considerou-se o coordenador na posição de R4 que está entre o auditório e a sala de exposições temporárias. Estes resultados encontram-se na Figura VIII.3.

VIII.1.5.Quinto teste

O quinto teste foi efectuado a 14 de Junho de 2010, sendo que neste realizou-se vários testes de forma a descobrir qual a melhor forma para efectuar a propagação do sinal do piso 1 para o piso 2, assim como, analisar o sinal que chega aos bloqueadores do piso 2. Os resultados são apresentados na Tabela VIII.4 e na Figura VIII.4. De seguida, mediu-se os vários bloqueadores no piso 0 situados no parque de estacionamento, cujos resultados podem ser observados na Figura VIII.1.

VIII.1.6.Sexto teste

O sexto teste foi realizado a 30 de Junho de 2010. Medimos a sala didáctica A e a sala didáctica B, no piso 2, a partir do router R5, no piso 1. O resultado é apresentado na Figura VIII.4. No final deste teste, começou-se a analisar o sinal a propagar-se do piso 0 para o piso -1.

VIII.1.7.Sétimo teste

O sétimo teste foi realizado a 5 de Julho de 2010. Analisamos várias posições para obter a melhor que nos permitisse passar o sinal para o piso -1. Mediu-se a sala de depósito de peças museológicas e a sala de depósito de material biológico. Também verificou-se que o sinal chega à câmara frigorífica, à câmara congeladora e aos bloqueadores. Os resultados encontram-se na Figura VIII.5.

VIII.1.8.Oitavo teste

O oitavo teste foi realizado dia 8 de Outubro de 2010. Este teste surgiu com a necessidade de efectuar a monitorizar de um cofre que se encontra na sala do segurança no parque de estacionamento do piso 0. O coordenador foi colocado no local do router R2, que podemos observar na Figura VIII.1. Verificamos como se propagava o sinal até a sala do segurança, como podemos visualizar na Figura VIII.7. Devido ao sinal não se propagar de forma ideal, foi decidido a colocação de mais um router, que se encontra exemplificado na Figura VIII.8.

VIII.1.9.Resultados Obtidos

Na Figura VIII.1 é apresentado os valores de *RSSI* e as respectivas posições dos dispositivos da rede, no piso 0.



Figura VIII.1: Medições no piso 0.

Na Figura VIII.1 os valores medidos na sala de cetáceos são em relação ao coordenador. Na sala de projecção de vídeo, sala caça da baleia, zona de passagem até ao posto de vigia os valores foram obtidos são em relação ao router R1.

Na sala caça da baleia, o valor lido dentro da vitrina sem vidro de -51dBm e com vidro de -60dBm, o dispositivo de teste foi colocado o mesmo local.

Os valores obtidos para o ponto A e B na sala de cetáceos encontram-se registados na Tabela VIII.1.

Tabela VIII.1 – Valores medidos nos pontos A e B na sala de cetáceos.

	PtoA (dBm)	PtoB (dBm)
Nó	-42	-44
Bloqueador	-50	-54

De acordo com a Tabela VIII.1, verifica-se que os bloqueadores têm pior recepção de sinal, pois não utilizam o *XBee Pro*.

Os testes realizados para determinar a localização do *router* R2 são apresentados na Tabela VIII.2.

Tabela VIII.2 – Testes realizados para determinar a localização do *router* R2.

	Pto1 (dBm)	Pto2 (dBm)	Pto3 (dBm)	Pto4 (dBm)	Pto5 (dBm)	Pto6 (dBm)
Do coordenador	-57	-57	-74	-66	-----	-----
Do ponto 4	-----	-----	-----	-----	-60	-57
Do ponto 2	-----	-----	-----	-----	-74	-75
Do ponto 3	-----	-----	-----	-----	-52	-51

Ao observar a Tabela VIII.2, verificamos que é necessário a colocação de um *router* na garagem de modo a que o sinal chegue aos pontos 5 e 6. Este *router*, denominado R2, é colocado no pagão, como é mostrado na Figura VIII.1.

Do piso 0 para o piso -1 existe um bloqueador localizado nas escadas, como podemos observar na Figura VIII.2. Realizou-se vários testes para passar o sinal do *router* R1 até ao bloqueador, sendo necessário introduzir os *routers* R9 e R10. O *router* R11 é para propagar o sinal até ao piso -1, através de uma zona técnica.

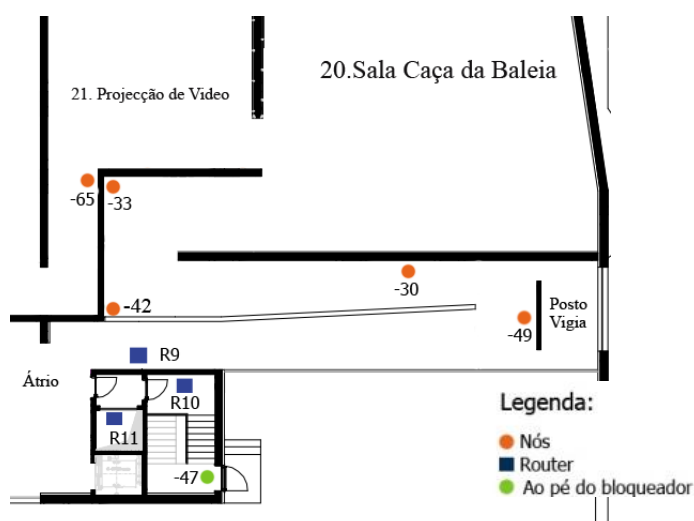


Figura VIII.2: Medições no piso 0.

Na Figura VIII.2, os valores a cor-de-laranja, são em relação a R9.

Os valores obtidos nos vários testes estão representados na Tabela VIII.3:

Tabela VIII.3: Valores obtidos entre os routers.

	R9 (dBm)	R11 (dBm)	R12 (dBm)	Bloqueador (dBm)
De R9	-----	-----	-----	-----
De R10	-47	-65	-----	-47
De R11	-----	-----	-45	-----

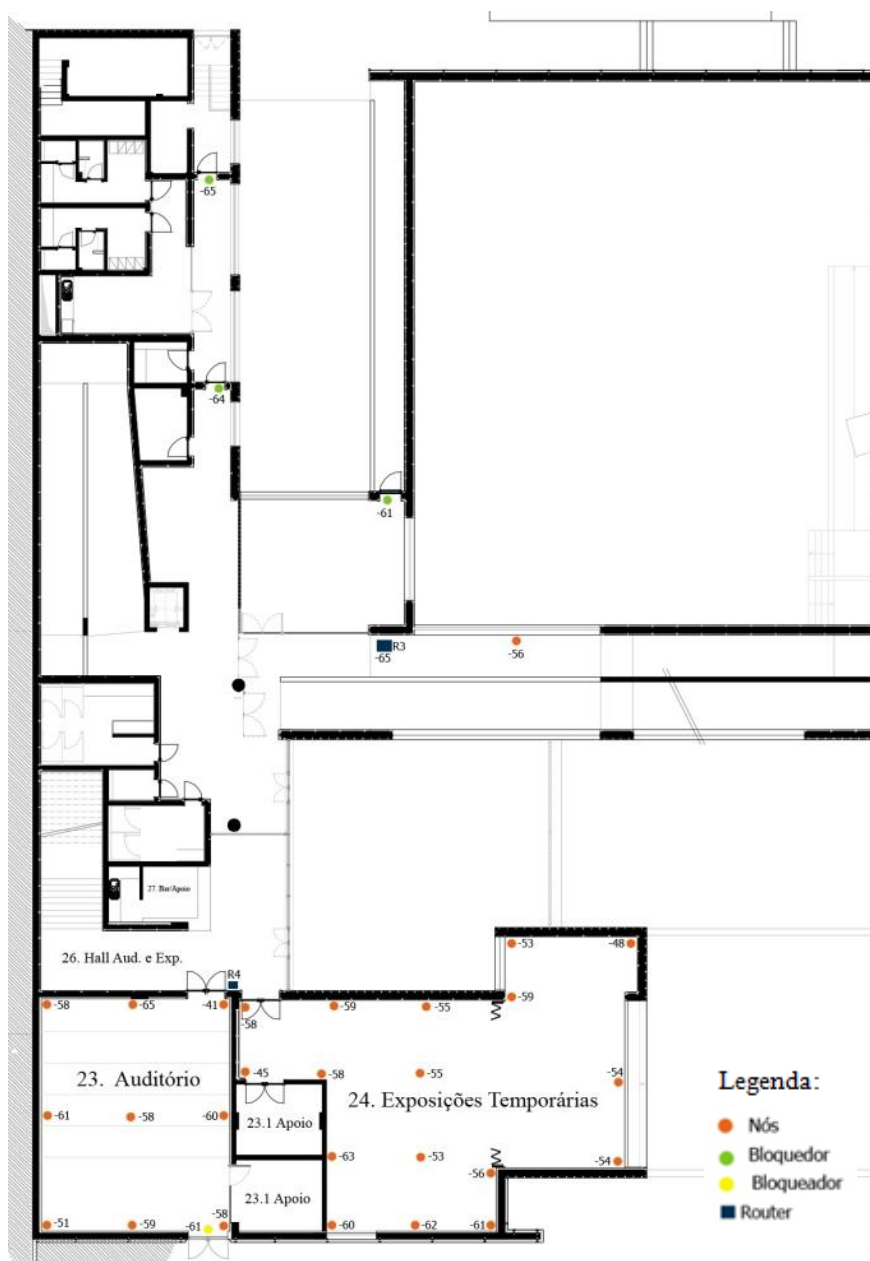


Figura VIII.3: Valores medidos no piso 1.

Na Figura VIII.3, na rampa onde se encontra o router R3, existe um valor de -56dBm medido em relação ao coordenador que fica no piso 0. Na posição do router R3 tem-se -65dBm medido também em relação ao coordenador. Verificou-se que o sinal de R3 para o R4 era de -56dBm.

Os valores medidos no auditório e na sala de exposições temporárias são em relação a R4. O bloqueador representado a amarelo no auditório também foi medido em relação a R4.

Os bloqueadores a verde foram medidos em relação a R3 e com as portas de emergência fechadas. Verificou-se que fechadas ou abertas estas portas não afectam muito a propagação do sinal, dado que existe vidros.

Os valores obtidos de RSSI para passar o sinal do piso 1 para o piso 2 estão representados na Tabela VIII.4.

Tabela VIII.4 – Valores medidos para obter a melhor localização de R5.

	Pagão1 no piso 2 (dBm)	R5 (dBm)	Área Didáctica B (dBm)	BloqA no piso 2 (dBm)	BloqB no piso 2 (dBm)
De R3	-61	-75	-----	-----	-53
Do Pagão1	-----	-----	-63	-----	-70
De R4	-60	-40	-----	-58	-64

Por análise à Tabela VIII.4 verifica-se a necessidade de um router no piso 2, o qual denominamos R5. Este dispositivo é colocado na entrada mais à direita da Área DidácticaA, como é apresentado na Figura VIII.4.

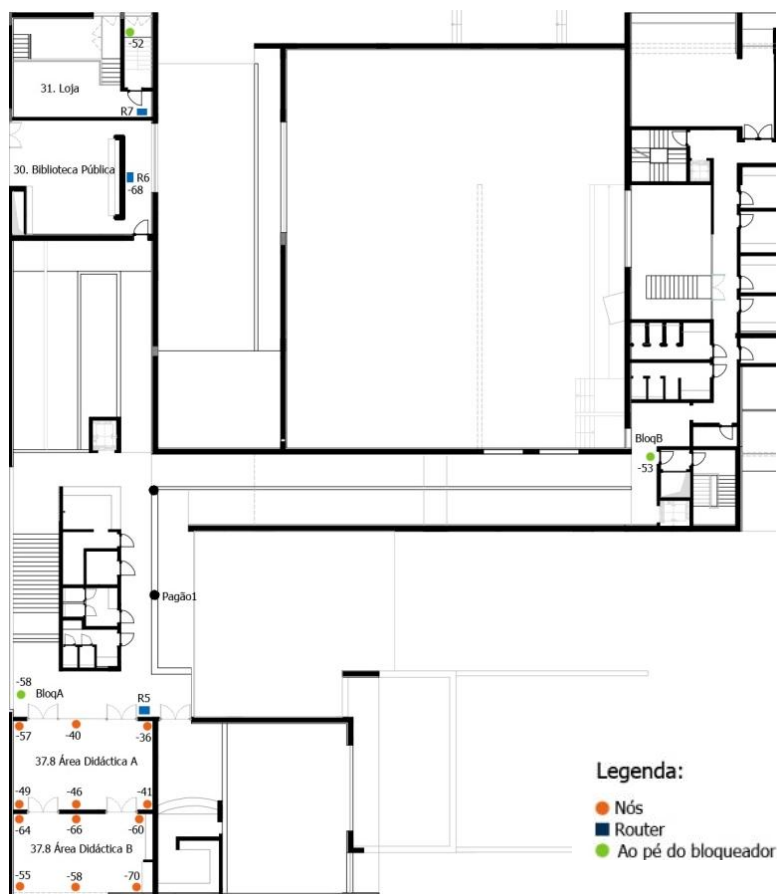


Figura VIII.4: Valores medidos no piso 2.

Na Figura VIII.4 os valores medidos nas áreas didáticas são em relação a R5. O valor medido no BloqA é em relação a R4 que encontra-se no piso 1. O valor obtido em R6 é em relação a R5. Verificou-se que o sinal de R6 para o bloqueador, localizado nas escadas, não chega. Assim, colocou-se o router R7 na loja e obteve-se o valor de -52dBm de R7 até ao bloqueador nas escadas.

Note-se que o teste realizado do pagão1 (ver Figura VIII.4) para a área didáctica B, o dispositivo de teste encontrava-se no canto inferior esquerdo desta sala.

Também foram efectuados testes no piso -1, sendo que a Tabela VIII.5 e a Figura VIII.5 mostram os resultados:

Tabela VIII.5 – Valores medidos no piso -1.

	Pto1 (dBm)	Pto2 (dBm)	Pto3 (dBm)	Pto4 (dBm)	Pto5 (dBm)
De R14	-57	-59	-73	-68	-----
De R15 (no ponto 4)	-65 (com XBee) e -39 (com XBee Pro)	-----	-47	-----	-64
De Pto2	-----	-----	-----	-----	-78

Pela Tabela VIII.5 verifica-se que é necessário um router, R15, para o sinal conseguir chegar ao bloqueador no ponto 5.

Dado que a monitorização das câmaras é outro projecto, apenas verificou-se o sinal que chega até lá. Neste caso fez-se o teste com os dois tipos de XBee.

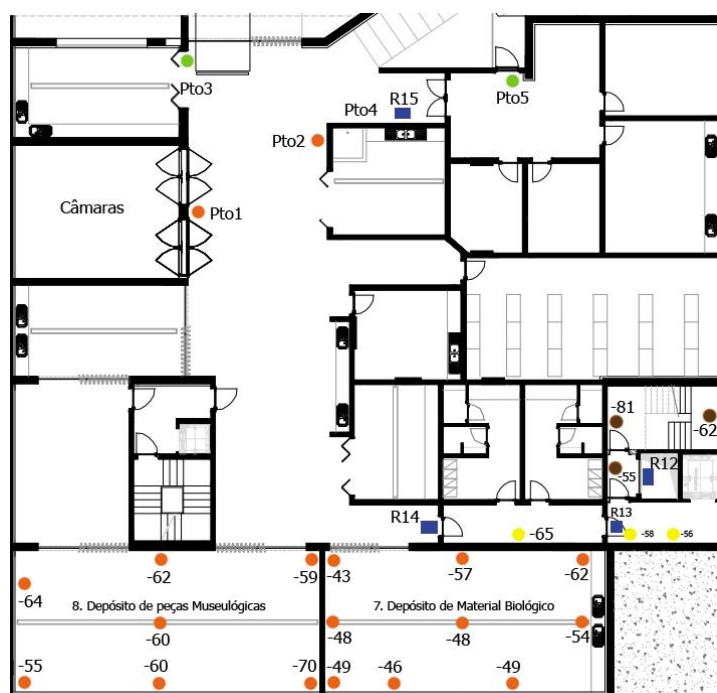


Figura VIII.5: Medições realizadas no piso -1.

Na Figura VIII.5, os nós a castanho são medidos em relação ao router R10 no piso 0. Os valores a amarelo são medidos em relação a R12 e os valores nos depósitos são medidos em relação ao R14. Verifica-se que para o sinal chegar aos depósitos são necessários três routers, R12, R13 e R14.

Na Figura VIII.6 encontra-se outros valores medidos no piso -1 para determinar a localização de R14.

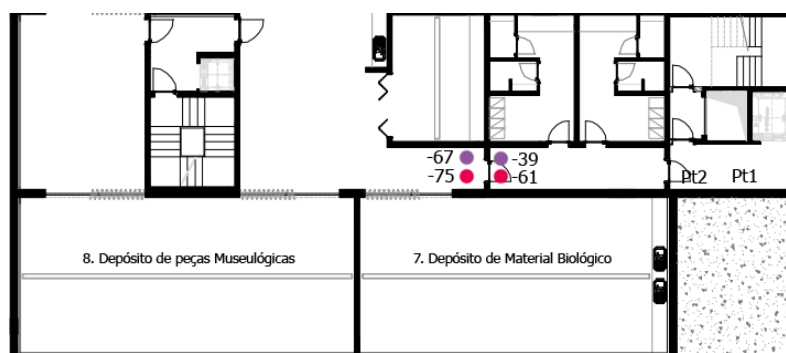


Figura VIII.6 – Medições realizadas no piso -1.

Pela Figura VIII.6 os nós a rosa são em relação ao ponto 1 (Pt1) e os nós a roxo são em relação ao ponto 2 (Pt2).

O último teste foi efectuado na zona do parque de estacionamento. O objectivo é monitorizar a sala do segurança. Para tal é necessário efectuar o teste de propagação entre o router R2, que se encontra mais perto da sala do segurança, e a sala do segurança. Os valores obtidos nestes testes encontram-se na Figura VIII.7.

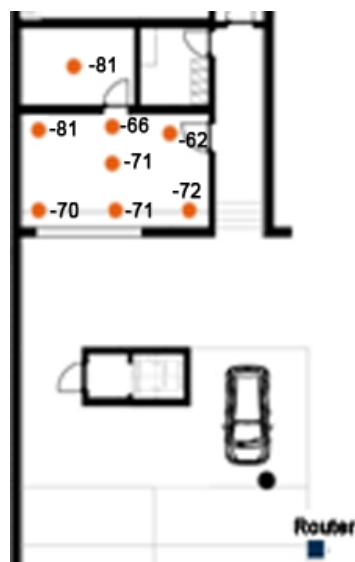


Figura VIII.7: Medições no piso 0, na zona da sala do segurança.

A sala a monitorizar tem o valor de *RSSI* de -81dBm. Devido a este valor alto, é necessário colocar um router entre o R2 e a sala. Desta forma, foram retirados alguns valores da sala próxima à da sala do segurança.

Ao analisar os valores retirados, foi decidido colocar o router no local apresentado na Figura VIII.8.



Figura VIII.8: Medições no piso 0, da sala do segurança até a sala do cofre.

VIII.1.10.Topologia proposta para a rede do Museu da Baleia

Após termos analisado a forma como o sinal propaga-se nas salas a monitorizar definiu-se os locais mas apropriados para colocar os nós sensores. Para tal teve-se em conta os seguintes aspectos:

- Valor do RSSI;
- Obstáculos na sala;
- Dimensões da sala;
- Tomada eléctrica.

A Figura VIII.9 apresenta a localização final dos nós sensores, bloqueadores e routers no piso 0.

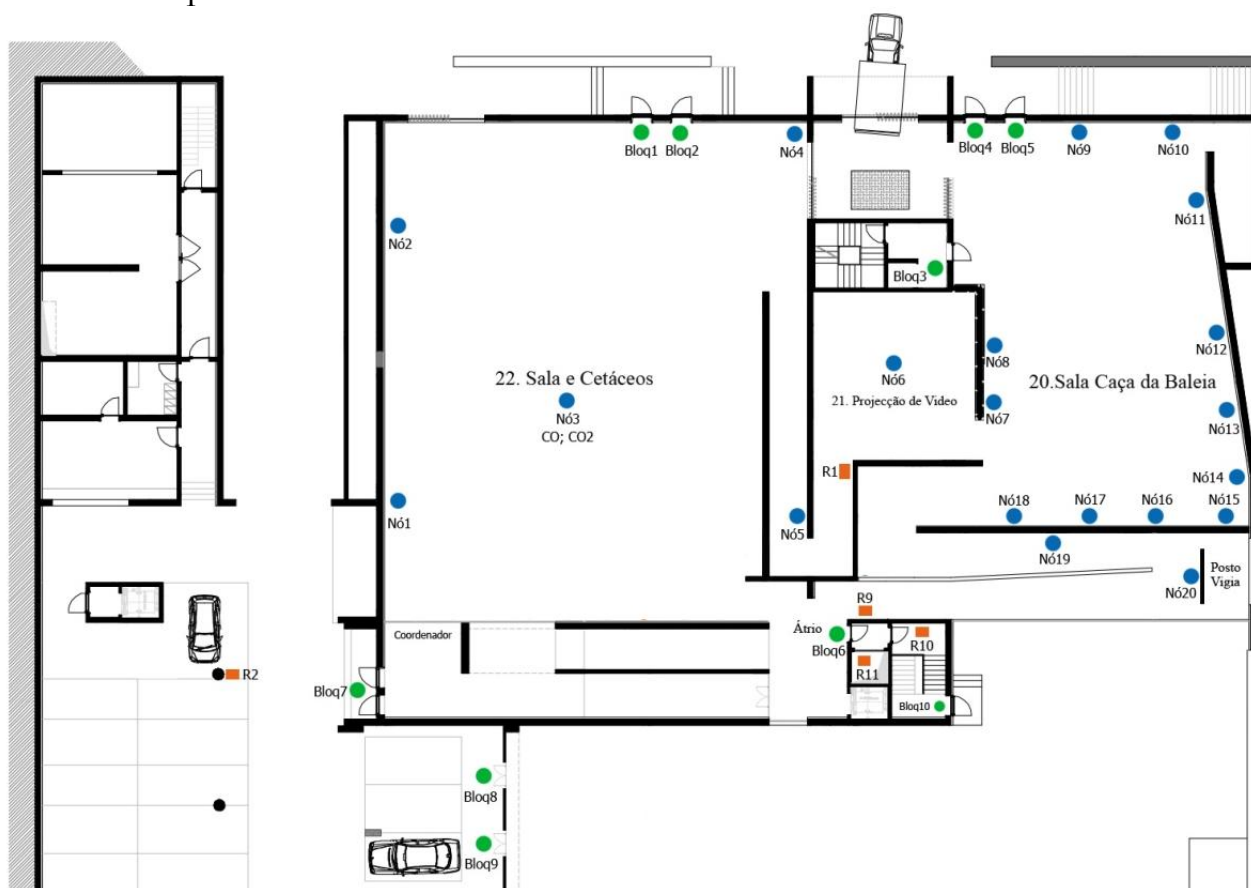


Figura VIII.9: Localização dos nós, bloqueadores e routers no piso 0

Neste piso é necessário: 20 nós; 10 bloqueadores e 5 routers.

As Figuras VIII.10 e VIII.11 apresentam as localizações dos dispositivos nos pisos 1 e 2 respectivamente.

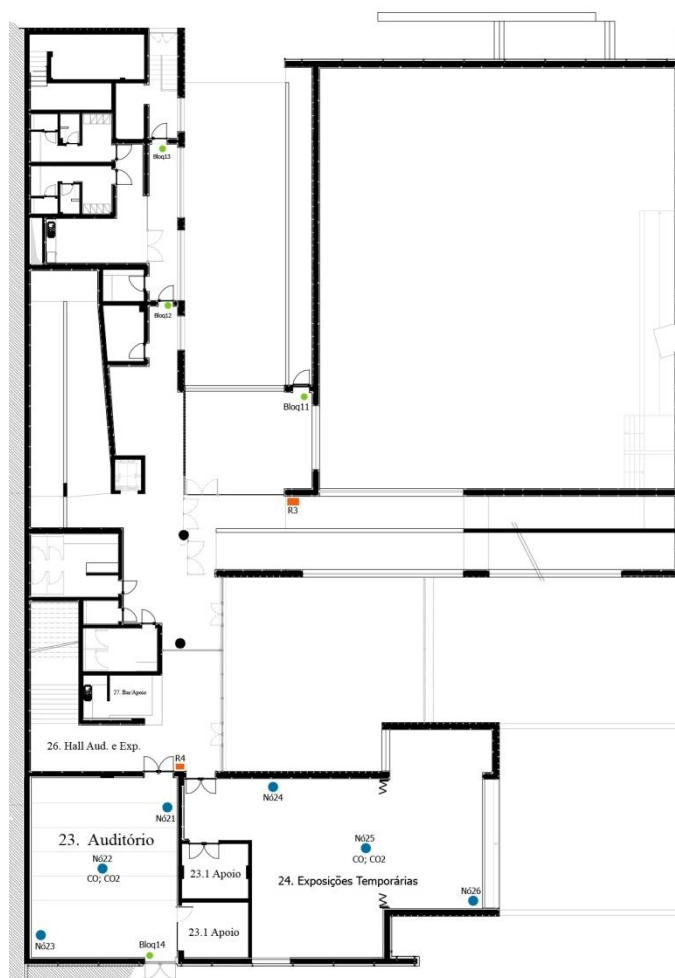


Figura VIII.10: Localização dos nós, bloqueadores e routers no piso 1.

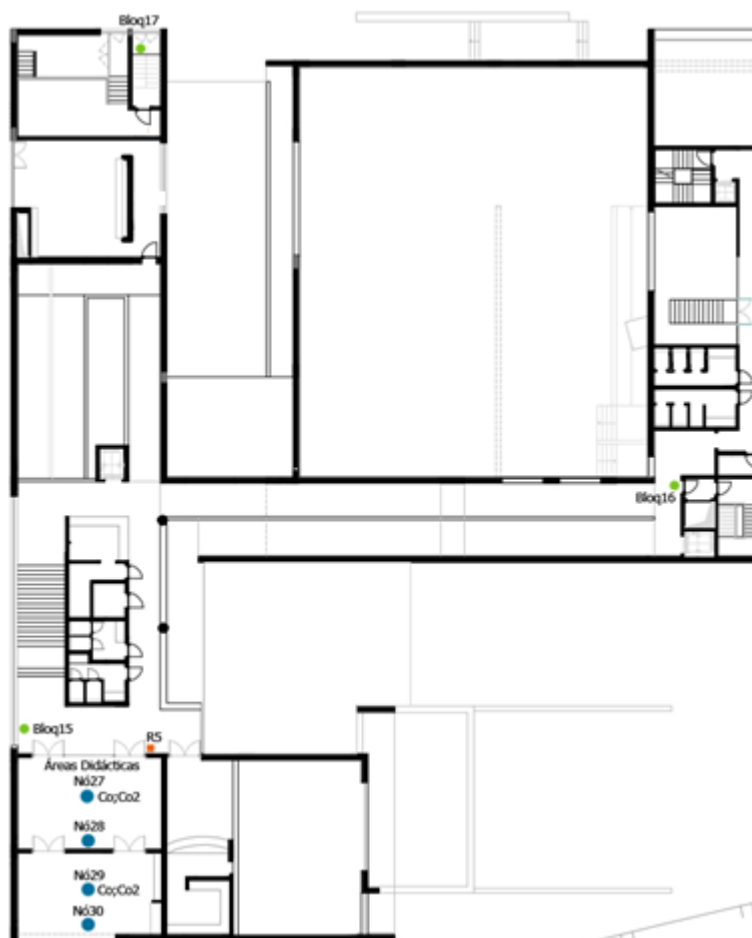


Figura VIII.11: Localização dos nós, bloqueadores e routers no piso 2.

No piso 1 tem-se 6 nós sensores, 2 routers e 4 bloqueadores.

De acordo com a figura 27 o piso 2 tem quatro nós sensores, 3 routers e 3 bloqueadores.

A Figura VIII.12 apresenta a localização dos dispositivos no piso -1.



Figura VIII.12: Localização dos nós, bloqueadores e routers no piso -1.

No piso -1 tem-se 2 nós sensores, 4 routers e dois bloqueadores.

Através deste estudo de propagação, será instalada uma rede de sensores sem fios no Museu da Baleia. Esta rede será composta por 1 coordenador, 19 bloqueadores, 16 routers e 37 nós sensores.

A escolha dos locais para os routers é crucial, visto que são utilizados para efectuar a comunicação entre o coordenador e os nós sensores.

Um dos problemas detectados neste estudo foi a variação do sinal medido numa mesma posição, devido às reflexões do sinal, à área onde foi realizada a monitorização, aos obstáculos e às interferências (pessoas circulando na área)

VIII.2. ANEXO B

Neste anexo encontra-se o um excerto do ficheiro *MS Word* criado ao exportar as amostras das sessões:

Report on: Tuesday 23 November 2010 19:40:55						
Teste 4						
Sessão monitoriza por fenómeno a monitorizar.						
Session active on: 16/11/2010 at 19:33 to 16/11/2010 at 19:37.						
Nodes of the session:						
No 2						
No 3						
No 5						
Phenomenons monitored:						
Temperature						
Humidity						
Light						
Session Samples						
Monitoring Data						
Date	Node	Temp	Hum	Light	Bat	
2010-11-16 19:34:02.0	2	18,98	71,58	67,14	2,60	
2010-11-16 19:34:08.0	3	19,01	73,70	06,71	2,68	
2010-11-16 19:34:12.0	2	18,94	71,67	47,00	2,58	
2010-11-16 19:34:18.0	3	18,98	73,99	20,14	2,67	
2010-11-16 19:34:22.0	2	18,93	71,75	60,42	2,60	
2010-11-16 19:34:28.0	3	18,96	74,19	13,43	2,68	
2010-11-16 19:34:32.0	2	18,93	71,75	60,42	2,60	
2010-11-16 19:34:38.0	3	18,94	74,33	13,43	2,68	
2010-11-16 19:34:42.0	2	18,85	71,81	53,71	2,57	
2010-11-16 19:34:48.0	3	18,92	74,72	13,43	2,68	
2010-11-16 19:34:52.0	2	18,88	72,01	06,71	2,60	
2010-11-16 19:34:58.0	3	18,89	74,44	13,43	2,68	
2010-11-16 19:35:02.0	2	18,88	72,04	06,71	2,59	
2010-11-16 19:35:08.0	3	18,84	74,19	06,71	2,68	
2010-11-16 19:35:12.0	2	18,84	72,10	06,71	2,56	
2010-11-16 19:35:18.0	3	18,80	74,02	06,71	2,67	
2010-11-16 19:35:28.0	3	18,78	74,02	06,71	2,68	
2010-11-16 19:35:32.0	2	18,81	72,24	06,71	2,56	

Figura VIII.13: Excerto do ficheiro gerado do histórico de uma sessão.

O ficheiro criado ao exportar o histórico das sessões, contém o identificador e a descrição da sessão, o que a sessão monitoriza e quais são os nós que monitorizam. Além disto, é apresentada uma tabela com as amostras recolhidas no intervalo de tempo escolhido pelo utilizador.