



**TECNOLOGIA
SETÚBAL**

ESCOLA SUPERIOR
POLITÉCNICO SETÚBAL

BERNARDO
MANUEL CAETANO
MILHEIRO

**CAMPUS DAS COISAS – UMA
ARQUITETURA IOT PARA O
CAMPUS DO IPS**

Relatório de Dissertação do Mestrado em
Engenharia Eletrotécnica e de Computadores

ORIENTADOR

Doutor Filipe Duarte dos Santos Cardoso

Dezembro 2024

BERNARDO
MANUEL CAETANO
MILHEIRO

**CAMPUS DAS COISAS – UMA
ARQUITETURA IOT PARA O
CAMPUS DO IPS**

JÚRI

*Presidente: Doutor Armando José Pinheiro Marques
Pires, ESTSetúbal/IPS*

*Orientador: Doutor Filipe Duarte dos Santos Cardoso,
ESTSetúbal/IPS*

*Arguente: Doutor António Paulo Duarte Gomes de
Abreu, ESTSetúbal/IPS*

Dezembro 2024

*Um agradecimento especial à minha família e a todos os meus amigos,
colegas e professores que comigo partilharam esta jornada académica.*

Agradecimentos

Em primeiro lugar, gostaria de agradecer a todas as pessoas com quem tive o enorme prazer de partilhar momentos inesquecíveis, dentro e fora do IPS, pela sua contribuição para o meu crescimento pessoal e profissional e conseqüentemente, para a conclusão desta dissertação e para o meu sucesso académico.

Em particular, gostaria de agradecer ao Professor Filipe Cardoso pela orientação, disponibilidade e acompanhamento constante ao longo do desenvolvimento desta dissertação e ainda, pelo seu contributo especial para o meu futuro académico e profissional.

Gostaria de agradecer aos Professores Manuel Ferreira, José Sousa, Jorge Martins e Luísa Caeiro pelo acompanhamento e por toda a contribuição no arranque e no decorrer deste trabalho e ainda, ao Engenheiro Rui Pereira da Logicalis, pela cedência inicial de material necessário ao desenvolvimento desta dissertação.

Uma palavra especial de agradecimento aos meus colegas e monitores de laboratório David Lopes, Eduardo Silva e João José pelo acompanhamento e colaboração ao longo do desenvolvimento desta dissertação, contribuindo sempre com os seus pontos de vista mais práticos e com a sua disponibilidade.

Um agradecimento especial à minha família por todo o apoio incondicional e por todo o contributo no meu percurso académico e profissional que continuo a construir.

Por fim, gostaria ainda de agradecer ao Instituto Politécnico de Setúbal e à Escola Superior de Tecnologia de Setúbal por me terem proporcionado uma experiência académica incrível e a todos os seus docentes que me acompanharam ao longo dos últimos anos.

Resumo

Através da realização desta dissertação, foi desenhada e implementada uma rede IoT no *campus* de Setúbal do Instituto Politécnico de Setúbal, com o objetivo de promover a pesquisa e o desenvolvimento de projetos IoT nesta instituição de ensino superior, constituindo desta forma uma base fundamental para a melhoria e atualidade da formação ministrada nesta área. A implementação desta rede permitirá a aplicação de diversos dispositivos e sensores ao longo do *campus*, transformando-o num *campus* mais inteligente e contribuindo para uma maior eficiência na utilização de recursos. Para além disso, permitirá ainda a toda a comunidade da área envolvente ao *campus* como empresas, escolas e outras entidades desenvolverem projetos em colaboração e com o apoio do Instituto Politécnico de Setúbal. Esta rede IoT baseia-se na comunicação de longo alcance LoRa e na utilização de servidores *cloud*, *open-source* e de carácter acessível e comunitário. De forma a testar o potencial e o alcance da rede, foi implementado um dispositivo comercial LoRa e desenvolvido um protótipo para testes de comunicação LoRa que permitiu construir um mapa de cobertura para o *campus* e para a zona circundante ao mesmo, tendo sido obtido um alcance máximo de 17,5 km nos testes realizados. Os resultados obtidos permitem concluir que a rede IoT implementada e que a tecnologia de comunicação utilizada abrangem todo o *campus* e ainda diversos locais onde se encontram empresas, instituições e organizações que poderão beneficiar das vantagens que a IoT oferece.

Palavras-chave: IoT, *Smart Campus*, IPS, LoRa, Sensores, Cobertura.

Abstract

Through this dissertation, an IoT network was designed and implemented on the Polytechnic Institute of Setúbal campus, aiming to promote the research and development of IoT projects at this higher education institution, thus constituting a fundamental basis for the improvement of up-to-date education provided in this area. The implementation of this network will also allow the implementation of various devices and sensors throughout campus, making it a smarter and more technological advanced campus, by contributing to a greater efficiency in the use of resources. It will also enable the entire community in the area surrounding the campus, such as companies, schools and other entities, to develop projects in collaboration with and with the support of the Polytechnic Institute of Setúbal. This IoT network is based on LoRa long-range communication and the use of cloud servers, which provide greater accessibility and are also open-source and community-based. In order to test the network's potential and range, a commercial LoRa device was implemented and a test prototype device was developed to allow the creation of a coverage map for the campus area and its surroundings, which shows a maximum range of 17,5 kilometers being obtained in the tests carried out. The obtained results show that the IoT network and the communication technology used in it can cover the entire campus, as well as various locations where there's various companies, institutions and organizations that could benefit from the advantages offered by IoT.

Keywords: IoT, Smart Campus, IPS, LoRa, Sensors, Coverage.

Índice

AGRADECIMENTOS.....	II
RESUMO	III
ABSTRACT.....	IV
ÍNDICE	V
LISTA DE FIGURAS	VII
LISTA DE TABELAS.....	IX
LISTA DE SIGLAS E ACRÓNIMOS	X
LISTA DE SÍMBOLOS	XIII
1. INTRODUÇÃO.....	1
2. ESTADO DA ARTE.....	5
2.1 Internet das Coisas.....	5
2.1.1 Aplicações IoT	8
2.1.2 Tecnologias de comunicação IoT	12
2.2 LoRa	19
2.3 <i>Link Budget</i> e RSSI	30
3. SMART CAMPUS.....	32
3.1 Conceito de <i>Smart Campus</i>	32
3.2 Casos de Estudo	34
4. DISPOSITIVOS E PLATAFORMAS LORA	45
4.1 Dispositivos LoRa	45
4.1.1 Módulos de comunicação LoRa	45
4.1.2 Nós Sensores LoRa.....	47
4.1.3 Gateways LoRa	50
4.2 Plataformas LoRa	54
4.2.1 Servidores de rede LoRa.....	54
4.2.2 Servidores aplicativos	57
5. PROPOSTA DE IMPLEMENTAÇÃO IoT – CAMPUS DE SETÚBAL – IPS	61
5.1 Planeamento da implementação IoT	61
5.2 Arquitetura da implementação IoT	66
5.3 Aplicações IoT para o <i>campus</i>	68
6. IMPLEMENTAÇÃO DA REDE IoT	70
6.1 Nós Sensores LoRa	70
6.1.1 Dispositivo para testes de cobertura – fieldTester (RFM95)	71
6.1.2 Sensor magnético de porta (Dragino LDS02)	75
6.2 Gateway LoRa	76
6.3 Servidor de rede LoRa	77
6.4 Servidor aplicativo IoT	87
7. TESTES DE COBERTURA.....	91
7.1 Medição da intensidade de sinal recebido	91
7.2 Recolha de dados.....	93
7.3 Mapa de cobertura	95
7.4 Análise de resultados	98

8. CONCLUSÃO E DESENVOLVIMENTOS FUTUROS.....	103
BIBLIOGRAFIA.....	106
ANEXO	116

Lista de Figuras

Figura 1: Mercado Global IoT (2020-2030) e Artigos de Pesquisa IoT (2013-2019) [7], [16].	6
Figura 2: Arquitetura de IoT [19].	6
Figura 3: Exemplos de aplicações IoT [23].	8
Figura 4: Exemplo de comunicação MQTT [11].	18
Figura 5: Comparação entre MQTT e CoAP [67].	19
Figura 6: Quota de mercado LoRa e LoRaWAN por setor de atividade (2022) [79].	20
Figura 7: Representação geral de uma rede LoRa [82].	22
Figura 8: Exemplo de modulação CSS [60].	23
Figura 9: Relação entre a modulação CSS e o fator de dispersão [61].	25
Figura 10: Camadas de comunicação da tecnologia LoRa [83].	26
Figura 11: Estrutura de uma trama LoRa [59].	27
Figura 12: Comportamento das classes de comunicação do LoRaWAN [86].	28
Figura 13: Componentes de <i>Link Budget</i> para transmissão de sinal.	30
Figura 14: Influência da reflexão na superfície terrestre no parâmetro n .	31
Figura 15: Componentes e Recursos IoT constituintes de um <i>campus</i> inteligente [88].	33
Figura 16: Distribuição dos nós sensores no espaço Tellus [90].	36
Figura 17: Página inicial da <i>dashboard</i> Tellus [90].	36
Figura 18: Integração de objetos digitais interativos no espaço Tellus [90].	37
Figura 19: Resultado da abordagem utilizada nas luminárias [90].	38
Figura 20: Aplicação móvel de controlo das luminárias [90].	38
Figura 21: Arquitetura IoT da Universidade de Coruña [91].	43
Figura 22: Variações dos níveis de CO ₂ ao longo do primeiro dia de exames [91].	43
Figura 23: Módulo de comunicação LoRa RFM95 [94].	47
Figura 24: Nó sensor ERS2 CO ₂ da Elsys [96].	48
Figura 25: Nó sensor Aranet4 PRO da Aranet [97].	49
Figura 26: Nó sensor LDS02 da Dragino [98].	50
Figura 27: <i>Gateway</i> Conduit IP67 Base Station da Multitech [99].	51
Figura 28: <i>Gateway</i> Aranet PRO Base Station [100].	52
Figura 29: <i>Gateway</i> Kerlink Wirnet iStation [101].	53
Figura 30: Visão geral da aplicação IPS-LoRa - TTSS.	56
Figura 31: <i>Dashboard</i> – Chirpstack [104].	56
Figura 32: Painel de controlo – Thingpark Enterprise [105].	57
Figura 33: Exemplo de <i>dashboard</i> – ThingWorx [106].	58
Figura 34: Exemplo de <i>dashboard</i> – Home Assistant [107].	58
Figura 35: Exemplo de <i>dashboard</i> – TagIO [108].	59
Figura 36: Painel de controlo – TagIO [108].	60
Figura 37: Mapa topográfico do <i>campus</i> de Setúbal do IPS [109].	62
Figura 38: Representação 3D do <i>campus</i> de Setúbal do IPS [110].	62
Figura 39: Mapa topográfico da região de Setúbal [109].	63
Figura 40: Exemplo de cobertura para um raio de 5 km do <i>campus</i> .	64
Figura 41: Exemplo de cobertura para um raio de 10 km do <i>campus</i> .	64
Figura 42: Arquitetura da implementação IoT no <i>campus</i> de Setúbal do IPS.	67
Figura 43: Esquemático e ligações do fieldTester.	71
Figura 44: Representação física do fieldTester.	72
Figura 45: Fluxograma do código desenvolvido para o fieldTester.	74
Figura 46: Local de instalação do Dragino LDS02.	75
Figura 47: Local de instalação da <i>gateway</i> .	76
Figura 48: Interface <i>web</i> de configuração da <i>gateway</i> .	77
Figura 49: <i>Dashboard</i> inicial - TTSS.	77

Figura 50: Registo da <i>gateway</i> Kerlink Wirnet iStation - TTSS.	78
Figura 51: Visão geral da <i>gateway</i> KerlinkWirnet - TTSS.	78
Figura 52: "Location" da <i>gateway</i> KerlinkWirnet - TTSS.	79
Figura 53: "API keys" da <i>gateway</i> KerlinkWirnet - TTSS.	79
Figura 54: Integração do Dragino LDS02 - TTSS.	80
Figura 55: Integração manual do fieldTester - TTSS.	81
Figura 56: "Live data" da aplicação IPS-LoRa - TTSS.	81
Figura 57: "Payload formatters" da aplicação IPS-LoRa - TTSS.	82
Figura 58: "Webhooks" da aplicação IPS-LoRa - TTSS.	83
Figura 59: Ligações da <i>gateway</i> KerlinkWirnet - TTN Mapper.	83
Figura 60: "API keys" da aplicação IPS-LoRa - TTSS.	84
Figura 61: "Device overview" do nó sensor Dragino LDS02 - TTSS.	84
Figura 62: Comunicação <i>downlink</i> em "Messaging" do nó sensor Dragino LDS02 - TTSS.	85
Figura 63: "Location" do nó sensor Dragino LDS02 - TTSS.	85
Figura 64: Formatação do <i>payload</i> do nó sensor Dragino LDS02 - TTSS.	86
Figura 65: Formatação do <i>payload</i> em "Payload formatters" do fieldTester - TTSS.	86
Figura 66: "Settings" do fieldTester - TTSS.	87
Figura 67: Criação de um dispositivo - TagIO.	88
Figura 68: "Data" do nó sensor Dragino LDS02 - TagIO.	88
Figura 69: <i>Dashboard</i> IPS-LoRa - TagIO.	89
Figura 70: Botão "Downlink" da <i>dashboard</i> IPS-LoRa – TagIO.	90
Figura 71: Fluxograma da aplicação móvel (RSSI_GPS).	92
Figura 72: Aplicação móvel RSSI_GPS.	94
Figura 73: Mapa de cobertura do <i>campus</i> de Setúbal do IPS.	97
Figura 74: Mapa de cobertura da zona de Setúbal e área circundante.	97
Figura 75: Distância máxima de comunicação obtida nos testes de cobertura.	99
Figura 76: Alcance máximo de 47,39 km para $n = 2,797$	100
Figura 77: Alcance máximo de 61,98 km para $n = 2,729$	101
Figura 78: Valores de RSSI para os locais de teste da zona de Setúbal e área circundante.	101

Lista de Tabelas

Tabela 1: Comparação entre tecnologias de comunicação IoT	16
Tabela 2: Efeito da variação de <i>BW</i> e <i>SF</i> na sensibilidade do recetor [59].....	25
Tabela 3: Chaves e IDs necessários para cada método de ativação [77].....	29
Tabela 4: Comparação entre os módulos de comunicação LoRa SX1262 e SX1276 [92], [93] ...	46
Tabela 5: Especificações dos nós sensores ERS2 CO2 e Aranet4 PRO [96], [97]	49
Tabela 6: Especificações do nó sensor LDS02 da Dragino [98]	50
Tabela 7: Especificações das <i>gateways</i> Multitech Conduit IP67 e Aranet PRO [99], [100]	52
Tabela 8: Especificações da <i>gateway</i> Kerlink Wirnet iStation [101].....	53
Tabela 9: Custos da implementação IoT no <i>campus</i> de Setúbal do IPS	68
Tabela 10: <i>Payload</i> de <i>uplink</i> do Dragino LDS02 [98].....	75
Tabela 11: Valores de RSSI para os locais de teste do <i>campus</i>	95
Tabela 12: Valores de RSSI para os locais de teste da zona de Setúbal e área circundante.....	96

Lista de Siglas e Acrónimos

3GPP	<i>3rd Generation Partnership Project</i>
4G / 5G	<i>4th / 5th Generation</i>
ABP	<i>Activation By Personalization</i>
ADR	<i>Adaptive Data Rate</i>
AES	<i>Advanced Encryption Standard</i>
API	<i>Application Programming Interface</i>
AR	<i>Augmented Reality</i>
BLE	<i>Bluetooth Low Energy</i>
CAGR	<i>Compound Annual Growth Rate</i>
CoAP	<i>Constrained Application Protocol</i>
CSS	<i>Chirp Spread Spectrum</i>
CRC	<i>Cyclic Redundancy Check</i>
DC	<i>Direct Current</i>
DEE	Departamento de Engenharia Eletrotécnica
DTLS	<i>Datagram Transport Layer Security</i>
ESCE	Escola Superior de Ciências Empresariais
ESE	Escola Superior de Educação
ESTS	Escola Superior de Tecnologia de Setúbal
EUI	<i>Extended Unique Identifier</i>
FEC	<i>Forward Error Correction</i>
FSK	<i>Frequency-Shift Keying</i>
GPS	<i>Global Positioning System</i>

GSM	<i>Global System for Mobile Communications</i>
HTTPS	<i>Hyper Text Transfer Protocol Secure</i>
IDC	<i>International Data Corporation</i>
IoT	<i>Internet das Coisas (Internet of Things)</i>
IP	<i>Ingress Protection</i>
IPS	<i>Instituto Politécnico de Setúbal</i>
ISM	<i>Industrial, Scientific and Medical</i>
LED	<i>Light-Emitting Diode</i>
LTE	<i>Long-Term Evolution</i>
LoRa	<i>Long Range</i>
LPWAN	<i>Low Power Wide Area Network</i>
MAC	<i>Media Access Control</i>
mmWave	<i>Milimeter Wave</i>
MQTT	<i>Message Queuing Telemetry Transport</i>
MU-MIMO	<i>Multi-User, Multiple-Input, Multiple-Output</i>
NAT	<i>Network Address Translation</i>
NB-IoT	<i>Narrowband IoT</i>
OFDMA	<i>Orthogonal Frequency-Division Multiple Access</i>
OTAA	<i>Over-The-Air-Activation</i>
P2P	<i>Peer-to-Peer</i>
PoE	<i>Power over Ethernet</i>
QoS	<i>Quality of Service</i>
REST	<i>Representational State Transfer</i>

RFID	<i>Radio Frequency Identification</i>
RSSI	<i>Received Signal Strength Indicator</i>
SLAAC	<i>Stateless Address Auto Configuration</i>
SNR	<i>Signal-to-Noise Ratio</i>
SPI	<i>Serial Peripheral Interface</i>
SSH	<i>Secure Shell</i>
TLS	<i>Transport Layer Security</i>
TTN	<i>The Things Network</i>
TTSS	<i>The Things Stack Sandbox</i>
UDP	<i>User Datagram Protocol</i>
UMTS	<i>Universal Mobile Telecommunications System</i>
URL	<i>Uniform Resource Locator</i>
US\$	<i>United States Dollar</i>
VLAN	<i>Virtual Local Area Network</i>
WiFi	<i>Wireless Fidelity</i>
WoS	<i>Web of Science</i>
WPA	<i>Wi-Fi Protected Access</i>

Lista de Símbolos

BW	Largura de banda
CR	Taxa de codificação
CO_2	Dióxido de carbono
L_0	Atenuação de propagação
L_{Rx}	Perdas na cablagem (receção)
L_{Tx}	Perdas na cablagem (emissão)
G_{Rx}	Ganho da antena de receção
G_{Tx}	Ganho da antena de emissão
P_{Rx}	Potência de sinal recebida no recetor
P_{Tx}	Potência de emissão
R_S	Taxa de transmissão de símbolos
SF	<i>Spreading Factor</i>
T_S	Tempo de símbolo

1. Introdução

O desenvolvimento de tecnologias e conceitos como a Internet das Coisas tem contribuído para a transformação dos processos e tarefas do quotidiano em homólogos mais acessíveis, sustentáveis e digitais. A necessidade dessa transformação tem tendencialmente crescido devido a fatores como a globalização, a consciencialização ambiental ou o desenvolvimento tecnológico que, com vista á melhoria do ambiente comum, promovem a partilha de informação e recursos através do desenvolvimento de dispositivos e tecnologias mais acessíveis e mais interligadas entre si [1].

Inicialmente, o conceito de Internet das Coisas ou *Internet of Things* (IoT) surgiu como um meio de comunicação entre dispositivos e objetos, nomeadamente, através de tecnologia RFID, com o intuito de interligar o mundo digital com o mundo físico e a Internet a uma componente de sensorização de parâmetros físicos reais [2]. Uma das primeiras definições de IoT refere que esta é “uma infraestrutura de rede global e dinâmica com capacidades de autoconfiguração, baseada em normas e protocolos de comunicação” [3], enquanto que outras se referem à IoT como “uma tecnologia que permite criar um mundo, no qual todos os objetos se encontram ligados á Internet e a comunicar uns com os outros, sem necessidade de intervenção humana” [4]. Apesar desta definição se ter alterado ao longo do tempo e à medida que a tecnologia evoluía, o objetivo principal de capacitar os dispositivos com algum tipo de sensorização e comunicação, sem que este processo necessite de intervenção humana, manteve-se inalterado. Atualmente, o conceito é ainda bastante abstrato e divergente, o que se traduz em inúmeras definições semelhantes para o mesmo conceito [5].

A divergência do conceito de IoT resulta numa enorme aplicabilidade que se estende a todos os setores de atividade e a todo o tipo de sistemas e dispositivos ligados à Internet e a redes IoT, sendo que os seus limites aplicacionais estão diretamente associados aos seus limites tecnológicos. As aplicações IoT resultam da recolha e análise de dados tipicamente provenientes da localização, monitorização, sensorização ou atuação de dispositivos interligados através de uma comunicação adequada às condições do ambiente de implementação. Quanto mais eficazes, eficientes e económicas forem a recolha, a análise e a comunicação destes dados, maior o número de possíveis aplicações e implementações de IoT. Um exemplo de aplicação IoT será o resultado funcional e interativo deste processo, adequado às necessidades específicas do tipo de aplicação e ao formato de interação e visualização de informação pretendido [6].

Quando o conceito de IoT foi definido, a tecnologia era, de forma geral, bastante mais limitada e dispendiosa do que a atual. Esta evolução deu origem a uma redução da complexidade geral de implementação de diversas tecnologias, contribuindo para uma maior acessibilidade, diversidade e integração das mesmas. Entre muitos outros, um exemplo desta evolução é a gestão de tráfego nas cidades, na qual os semáforos e a iluminação pública, outrora operados de forma independente e manual, evoluíram para um controlo muito mais independente e adaptado em tempo real às necessidades da via pública e do tráfego existente [7].

O conjunto de tecnologias IoT que mais beneficiou da evolução tecnológica foram as tecnologias de comunicação sem fios, sendo estas um fator crucial para muitas aplicações e sistemas IoT que outrora não seriam possíveis de implementar. Atualmente, a diversidade de tecnologias de comunicação permite uma maior adequação da tecnologia escolhida ao tipo de implementação pretendido, sendo que o leque de possibilidades se estende desde tecnologias com enormes capacidades de transmissão como o WiFi 6 (*Wireless Fidelity 6*) ou o 5G (*5th Generation*), a tecnologias de baixo consumo energético como o Zigbee ou BLE (*Bluetooth Low Energy*) e ainda a tecnologias de comunicação de longo alcance como o NB-IoT (Narrowband-IoT) ou o LoRa (*Long Range*) [8]. A tecnologia de comunicação LoRa é uma das principais tecnologias de comunicação utilizadas em IoT devido ao seu alcance de comunicação (cobertura), baixo consumo energético e baixo custo de implementação, o que permite criar extensas redes de dispositivos com autonomia suficiente para alguns anos de utilização [9]. Os protocolos de comunicação evoluíram igualmente em conjunto com as tecnologias de comunicação e são fundamentais para a uniformização da comunicação em redes IoT, garantindo tipicamente uma comunicação mais eficiente e segura [10]. Entre os diversos protocolos de comunicação existentes, o protocolo MQTT (*Message Queuing Telemetry Transport*) é um dos que mais se destaca para implementações IoT devido à sua simplicidade de utilização e baixo consumo de recursos computacionais, adequando-se à limitação de recursos tipicamente associada a redes de sensores e algumas redes IoT. O MQTT opera sobre um modelo de publicar/assubcrever (*publish/subscribe*), que difere dos modelos tradicionais dos protocolos Internet por se focar numa comunicação tipicamente *broadcast* [11].

Uma análise mais aprofundada da adequação de uma tecnologia de comunicação a um tipo específico de implementação pode consistir na avaliação da qualidade e do alcance da comunicação naquele ambiente de propagação de sinal. Esta avaliação pode ser realizada através do *Link Budget* ou com base no RSSI (*Received Signal Strength Indicator*), determinando os valores associados à intensidade de sinal recebido e analisando os resultados obtidos.

Através destas tecnologias e protocolos de comunicação tem sido possível criar inúmeras redes de sensores, constituídas pelas mais variadas formas de comunicação e pelos mais variados tipos de sensores, sendo que as características de implementação e o tipo de aplicação são os principais fatores de distinção entre estas redes. Entre outras características, o ambiente de implementação, os recursos disponíveis e o tamanho da rede, fazem parte daquelas que irão determinar o tipo de dispositivos, plataformas e tecnologia de comunicação a utilizar. Uma rede de sensores sem fios é um tipo de rede IoT tipicamente constituída por dispositivos sensoriais ou nós sensores que podem comunicar entre si ou diretamente com o único dispositivo da rede ligado à Internet (*gateway*) e no qual toda a comunicação de dados converge. Para além dos nós sensores e da *gateway*, existem ainda o servidor de rede, responsável pela gestão da rede e o servidor aplicacional, responsável pela visualização e atuação sobre os dados no formato mais adequado ao tipo de aplicação pretendido. Estes servidores são também conhecidos como plataformas IoT e podem ser implementados de forma local ou acedidos através de aplicações *web* e servidores *cloud* [12].

As tecnologias IoT são cada vez mais uma componente crucial da criação de espaços mais inteligentes e sustentáveis, dotando-os de características tecnológicas e serviços inovadores. Por esse motivo, torna-se cada vez mais comum deparar-nos com sistemas IoT que impactam positivamente a comunidade onde estão inseridos, particularmente nos serviços disponíveis à população [13]. Um desses serviços é a educação, onde o conceito de *smart campus* e o impacto que este tem na educação e na sua comunidade, tem justificado a crescente adesão a estes sistemas. Através da utilização de tecnologias IoT, o conceito de *smart campus* visa transformar ambientes educativos tradicionais em ecossistemas inteligentes, com o intuito de otimizar o processo de aprendizagem e simultaneamente, promover a sustentabilidade da instituição e das suas infraestruturas. Este conceito tem sido aplicado de diversas formas e em diversos *campus* escolares, o que tem resultado em melhorias significativas no dia-a-dia da comunidade estudantil [14].

A gestão inteligente de recursos e a automação de tarefas são apenas algumas das características dos *smart campi* que resultam numa redução de custos, numa maior sustentabilidade, numa maior eficiência operacional e na melhoria generalizada da experiência académica dos alunos [15]. A crescente simplicidade de implementação do tipo de redes presentes em *smart campus*, juntamente com todas as oportunidades de investigação e inovação que uma rede IoT possibilita, constituem a motivação necessária para um estudo mais aprofundado do conceito de *smart campus* e das suas componentes IoT.

O desenvolvimento desta dissertação consistiu no desenho e implementação de uma rede IoT baseada em comunicação LoRa no *campus* de Setúbal do Instituto Politécnico de Setúbal (IPS). A rede implementada é composta por dois nós sensores com funções e configurações distintas com o intuito de demonstrar as aplicabilidades e alcance da rede. Os servidores utilizados representam também um papel importante nesta implementação por possuírem uma componente *open-source* e comunitária e promoverem a acessibilidade necessária à pesquisa e desenvolvimento de projetos IoT. Foi ainda desenvolvido um método de recolha de dados que permitiu criar mapas de cobertura da rede e analisar os resultados obtidos ao longo do *campus* e da área de Setúbal.

O conteúdo desta dissertação divide-se em 8 capítulos, sendo que estes se encontram estruturados da seguinte forma: No Capítulo 1, é realizada uma introdução ao tema, ao trabalho realizado e à estrutura desta dissertação. No Capítulo 2, são estudados e enquadrados os conceitos abordados por esta dissertação, nomeadamente, o conceito de IoT, juntamente com as suas aplicações e tecnologias de comunicação, a tecnologia de comunicação LoRa e os conceitos de *Link Budget* e RSSI. No Capítulo 3, é apresentado o conceito de *smart campus* e de forma a aferir as diferentes aplicações, tecnologias e processos utilizados, serão analisados casos reais de implementação IoT em alguns *campus* universitários. No Capítulo 4, são apresentadas as diversas componentes que constituem uma rede LoRa e aprofundados alguns exemplos disponíveis no mercado, juntamente com as arquiteturas, dispositivos e plataformas mencionadas nos casos de estudo. No Capítulo 5, é apresentada uma proposta fundamentada de implementação de uma rede IoT para o *campus* de Setúbal do Instituto Politécnico de Setúbal, que permite enriquecer a sua qualidade de ensino e acrescentar valor tecnológico às infraestruturas do mesmo, tendo como principal objetivo, a inovação e o desenvolvimento de aplicações IoT em contexto educativo. Esta proposta é composta por uma caracterização do *campus* e da área circundante potencialmente abrangida pela rede LoRa, pela arquitetura de implementação da rede IoT proposta e por alguns exemplos de aplicações específicas para o *campus*. No Capítulo 6, é apresentado o processo de implementação proposto, iniciado pelo desenvolvimento do dispositivo de testes de cobertura e pela instalação do sensor comercial LoRa, criando posteriormente a rede LoRa através da instalação e configuração da *gateway* e do servidor da rede *The Things Stack Sandbox* (TTSS) e integrando os dados recolhidos no servidor aplicacional TagoIO. No Capítulo 7, é realizado o desenvolvimento de uma aplicação móvel com a capacidade de recolher os valores de RSSI do dispositivo de testes e associá-los a coordenadas de localização. Esses valores serão recolhidos em diversos locais do *campus* e da zona de Setúbal, de forma a criar mapas de cobertura e a permitir analisar o alcance da rede LoRa implementada. Por último, no Capítulo 8, é realizada uma reflexão final em forma de conclusão, onde são também apresentadas possíveis melhorias e desenvolvimentos futuros da rede IoT implementada.

2. Estado da Arte

Neste capítulo são apresentados os conceitos e tecnologias abordadas ao longo da dissertação, de forma a fornecer um enquadramento teórico e uma visão da aplicabilidade destes conceitos, em particular no contexto de *smart campus*. Este capítulo é composto pela análise detalhada do conceito de Internet das Coisas, juntamente com as suas aplicações e tecnologias de comunicação, com especial ênfase na tecnologia de comunicação LoRa e na obtenção de valores de *Link Budget* e RSSI.

2.1 Internet das Coisas

A IoT tem evoluído significativamente nos últimos anos devido aos avanços tecnológicos que têm permitido transformar a forma como os dispositivos e os sistemas interagem e comunicam entre si. Esta evolução permitiu reduzir os custos de implementação e simultaneamente, aumentar a capacidade de processamento e armazenamento dos dispositivos e plataformas, aumentando ainda a eficiência de todas as tecnologias envolvidas, particularmente das tecnologias de comunicação. A diversidade de tecnologias de comunicação possibilita o desenvolvimento de aplicações IoT mais específicas e mais eficientes, através da escolha das tecnologias mais adequadas às características da implementação em causa, permitindo reduzir os custos, os consumos e a complexidade da mesma. Estas melhorias abriram espaço para um maior número e diversidade de aplicações IoT que atualmente abrangem todo o tipo de tarefas, sendo que o desenvolvimento de plataformas IoT mais interativas e intuitivas contribuiu também para o crescimento das redes IoT existentes e para o desenvolvimento de novas redes, através de uma maior e melhor capacidade de gestão de dados [7], [16].

De acordo com a “International Data Corporation” (IDC), o mercado global de soluções IoT tem observado um crescimento anual de 20%, tendo em 2024, um valor estimado de 1,110 bilhões de US\$. Este valor irá continuar a crescer devido a todas as vantagens que a IoT oferece e ao potencial de crescimento exponencial que o investimento em IoT agrega, visto que o desenvolvimento de novos dispositivos, tecnologias e plataformas IoT torna a sua implementação mais acessível e essa acessibilidade cria mais oportunidades de desenvolvimento [16]. O aumento do número de artigos de pesquisa IoT disponíveis na base de dados da “Web of Science” (WoS) é outra prova do crescimento e adesão ao conceito, visto que esse número aumentou de 187 artigos em 2013, para 1436 em 2019 [7]. Na Figura 1 é possível observar o crescimento do mercado global de IoT entre 2020 e 2030 e o aumento do número de artigos de pesquisa IoT entre 2013 e 2019.

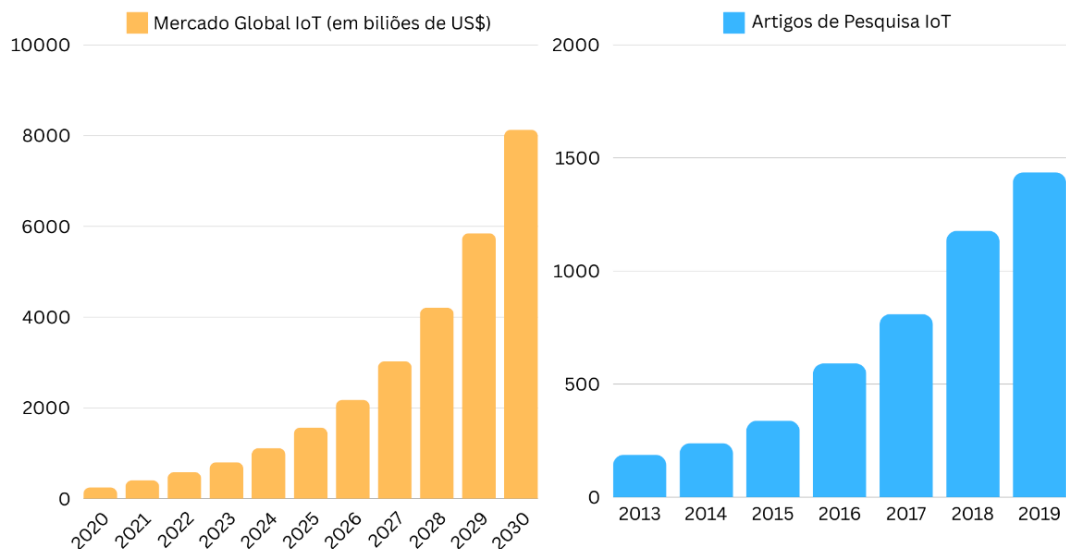


Figura 1: Mercado Global IoT (2020-2030) e Artigos de Pesquisa IoT (2013-2019) [7], [16].

A IoT é geralmente caracterizada por uma rede de “coisas” interligadas entre si, tipicamente com algum tipo de inteligência associada [17]. Estas “coisas” são qualquer dispositivo que, sem intervenção manual, tenha a capacidade de recolha e transmissão de dados e que por via de tecnologias de comunicação, se encontra interligado e a comunicar com as restantes “coisas” [18]. Esta comunicação na rede permite ainda alterar remotamente o comportamento das “coisas”, conforme o seu estado ou de acordo com os comportamentos das restantes “coisas” presentes na rede. Representada na Figura 2, uma das arquiteturas mais básicas de IoT é constituída por três camadas: a camada de aplicação, a camada da comunicação e a camada da sensorização [19].



Figura 2: Arquitetura de IoT [19].

A camada de aplicação é representada pelas diferentes aplicabilidades de IoT que, devido à sua abrangência e capacidade de adaptação, tem um enorme potencial de impacto nas mais diversas áreas sociais, ambientais e económicas. A IoT é aplicável a praticamente todos os setores de atividade e a recolha e análise de dados que lhe é característica, tem como principal objetivo a automatização e otimização de tarefas. Essa automatização é conseguida através da recolha e análise dos dados produzidos por essa tarefa ou aplicação e pela programação do seu processo de realização, de acordo com os dados recolhidos em tempo real ou em função dos resultados obtidos da análise dos mesmos [20].

A camada de comunicação é representada pelas diferentes tecnologias de comunicação utilizadas em redes IoT, seja para ligação ao exterior, como é o caso da Internet, seja para comunicação entre dispositivos na rede. As diversas tecnologias de comunicação diferenciam-se pelas suas características técnicas como a capacidade de transmissão, a latência, o alcance, a segurança, os custos de implementação e o consumo energético, sendo que cada uma destas se adequa melhor a diferentes cenários de aplicação. De acordo com as características da implementação, poderá ser possível utilizar tecnologias como o WiFi ou as comunicações móveis (4G/5G), que tipicamente oferecem uma capacidade de transmissão superior, ou então ser necessária uma tecnologia com menor consumo energético, como o *Bluetooth* ou o LoRa [8]. Devido à diversidade de características e ao número de tecnologias de comunicação disponíveis, um planeamento de implementação IoT exige que se realize uma análise prévia a estas tecnologias e à sua adequação ao cenário de implementação.

A camada de sensorização é representada pelos dispositivos que permitem recolher dados sobre fenómenos físicos, sendo que alguns permitem ainda atuar sobre os mesmos, de acordo com a finalidade da aplicação. Consoante a sua função, os dispositivos utilizados denominam-se de sensores ou atuadores, sendo que os primeiros são cada vez mais uma componente fulcral dos sistemas IoT. A importância da sensorização deve-se à necessidade de monitorização de diversos parâmetros físicos e à relevância dos resultados obtidos, os quais têm ajudado a aumentar a produtividade, a segurança, a gestão de recursos, entre outros fatores [21]. Enquanto o conceito de IoT representa um ecossistema mais amplo e integra a ligação à Internet com todo o tipo de dispositivos, tecnologias de comunicação e plataformas, o conceito e tipo de rede IoT que melhor define a arquitetura e o fluxo de comunicação entre os sensores e as restantes componentes necessárias à recolha e análise de dados, denomina-se de redes de sensores.

2.1.1 Aplicações IoT

Uma aplicação IoT é tipicamente descrita pela sua principal funcionalidade e pelos benefícios que a mesma fornece aos seus utilizadores, de forma a tornar a sua explicação e apresentação mais simplificada. No entanto, as aplicações IoT são o resultado palpável e interativo de um conjunto de componentes técnicas, apresentadas através de plataformas e interfaces desenhadas especialmente para este tipo de aplicações. Estas plataformas são responsáveis por fazer a ligação entre a rede e o utilizador, facilitando a interação, a análise e visualização dos dados. Estes dados podem ser apresentados de forma básica, no caso da aplicação se basear numa recolha simples de valores de temperatura ou humidade ou então de forma mais complexa, envolvendo análise e atuação em tempo real, como no caso dos sistemas de gestão de tráfego. Em ambos os casos, existem plataformas e interfaces que gerem e processam os dados recolhidos, para que estes sejam apresentados da forma mais relevante ao utilizador e de acordo com a sua aplicação [22].

Na Figura 3 é possível observar alguns exemplos de aplicações IoT associados a áreas de atividade, onde foram desenvolvidos conceitos, tecnologias e dispositivos inteligentes coincidentes com diferentes características de IoT, como a monitorização em tempo real ou a melhoria na tomada de decisões, sendo que estas duas características são comuns a muitos sistemas IoT atualmente [23].

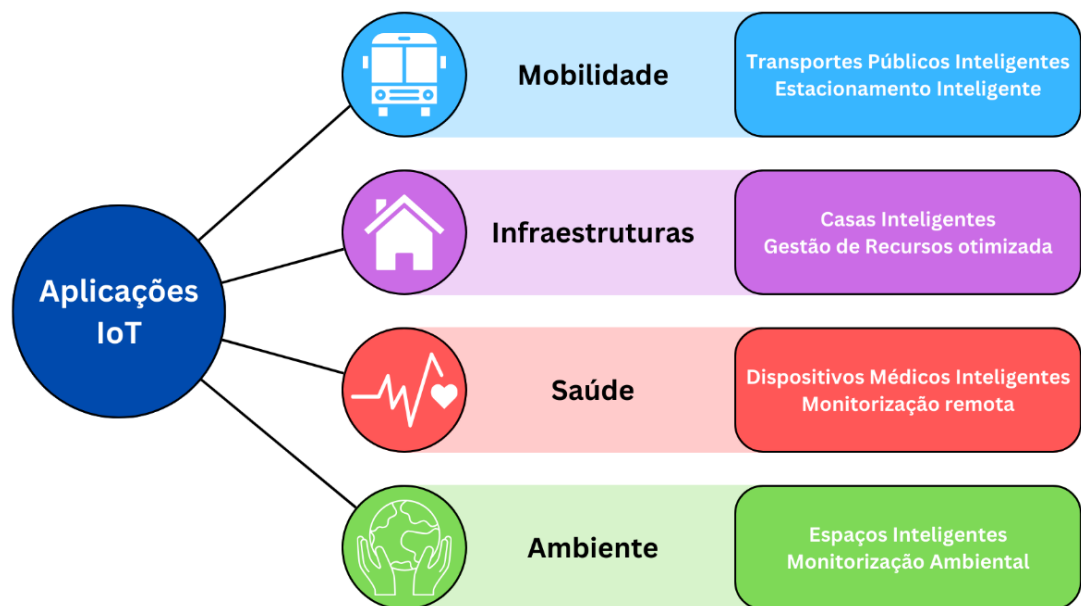


Figura 3: Exemplos de aplicações IoT [23].

A introdução de IoT na área da mobilidade permitiu criar um ambiente interligado entre os diversos meios de transporte e a própria via pública, nomeadamente com a implementação de sensores nos veículos e nas infraestruturas públicas, dando origem a sistemas de navegação mais otimizados, à partilha de informação em tempo real e à redução do consumo de recursos e do número de acidentes rodoviários. A partilha de informação referida deu origem a conceitos como os “Transportes Públicos Inteligentes” e o “Estacionamento Inteligente” [24].

Os transportes públicos inteligentes, como o nome indica, são transportes públicos equipados com algum tipo de inteligência e capacidade de recolher e transmitir informações sobre eles próprios e sobre a via pública onde se deslocam. As principais vantagens desta recolha e transmissão de dados são a possibilidade de otimização das rotas e a partilha, em tempo real, da localização dos transportes, o que permite informar os utilizadores de possíveis atrasos ou congestionamentos no trânsito [25].

O estacionamento inteligente refere-se aos parques e aos lugares de estacionamento disponíveis na via pública que, de alguma forma, conseguem transmitir o seu estado de disponibilidade aos utilizadores. Através de câmaras ou sensores, é possível desenvolver estacionamentos mais otimizados no que diz respeito à disponibilidade, capacidade e tempo de procura [26].

Num contexto de *smart campus*, estes conceitos de mobilidade inteligente podem facilmente ser implementados e adaptados às especificações próprias da comunidade estudantil. Num *campus* de grandes dimensões, por exemplo, é possível que exista um sistema interno de transportes para que a comunidade estudantil se desloque de forma mais cômoda ao longo deste. Nestes casos, a implementação de transportes inteligentes e as vantagens consequentes da mesma, tornam a utilização deste tipo de transportes mais atrativa. Os lugares de estacionamento, dentro e fora do *campus*, são uma componente da via pública bastante utilizada pela comunidade estudantil e, por esse motivo, a implementação de sensores e a criação de lugares de estacionamento inteligentes, torna a tarefa de estacionamento mais facilitada e organizada.

A introdução de IoT na área das infraestruturas permitiu desenvolver sistemas inteligentes com capacidade de monitorizar e controlar todas as componentes de uma infraestrutura, de forma a tornar os processos e o próprio ambiente, dentro e fora da infraestrutura, mais eficientes, eficazes e sustentáveis. Os diferentes parâmetros que constituem a gestão de uma infraestrutura como o aquecimento, a qualidade do ar, a ventilação, a luminosidade ou a gestão da água podem ser monitorizados e controlados de forma a aumentar o conforto dos presentes ou a reduzir o consumo de recursos [27].

Aplicada à habitação, a gestão destes parâmetros dá origem a conceitos como “Casas Inteligentes” ou “*Smart Homes*”. O objetivo principal deste conceito é a automação de processos domésticos e o controlo de diferentes partes da casa de forma mais inteligente e maioritariamente remota, através da utilização de diversas tecnologias de comunicação e sensorização. Assim, estas infraestruturas podem ajustar-se de forma autónoma e em tempo real às necessidades do habitante e às condições do seu próprio ecossistema. Um exemplo geral de aplicação deste conceito resume-se a um sensor que monitorize um certo parâmetro em tempo real e que permita identificar possíveis oportunidades de economia de recursos, como um sensor de luminosidade ligado a um sistema de iluminação ou como um sensor de humidade ligado a um sistema de rega [28].

As casas e infraestruturas inteligentes encontram-se sempre associadas a uma gestão de recursos mais otimizada devido à recolha e análise de dados mencionada, o que permite obter uma visão e um controlo mais alargado sobre a distribuição e o consumo desses recursos. Essa visão permite ajustar ou melhorar certos processos para que estes se tornem mais sustentáveis e eficazes no seu propósito [27].

Num contexto de *smart campus*, estes conceitos de infraestruturas inteligentes podem ser aplicados de forma a obter uma melhor gestão do número elevado de recursos, que naturalmente, um *campus* consome. Sendo a água e a energia, os consumos mais relevantes nas despesas de um *campus*, é importante que estes recursos sejam utilizados da forma mais eficiente e eficaz possível. A monitorização do consumo de água, por exemplo, permite detetar e atuar sobre possíveis fugas ou consumos inadequados em qualquer parte da infraestruturã. De forma semelhante, a monitorização do consumo energético permite também detetar o consumo inadequado por parte de dispositivos, máquinas ou iluminação.

A introdução de IoT na área da saúde permitiu transformar os habituais aparelhos médicos em sistemas e dispositivos inteligentes capazes de monitorizar os pacientes de forma mais cómoda, mantendo uma eficácia elevada na deteção e na análise de todo o tipo de eventos e alterações na saúde e comportamento dos mesmos. Estes dispositivos permitem recolher dados em tempo real sobre os diferentes parâmetros que permitem avaliar o estado de saúde atual do paciente como o seu batimento cardíaco, a glicose no sangue, a oxigenação, entre outros parâmetros e ainda possivelmente atuar sobre os mesmos. A utilização destes dispositivos serve sobretudo como forma de prevenção ou deteção antecipada de doenças ou condições e como forma de socorro mais eficaz e célere, sem a necessidade de os pacientes permanecerem constantemente em ambiente hospitalar, aumentando assim a sua qualidade de vida [29].

Estes dispositivos médicos inteligentes possuem diversos formatos e atualmente, em virtude da evolução tecnológica e da redução do tamanho dos sensores, é possível que um paciente transporte um destes dispositivos consigo no seu dia-a-dia e que através de monitorização remota, se saiba o seu estado de saúde a qualquer momento e em qualquer lugar [30]. Exemplo disto são os dispositivos *wearables* como os *smartwatches*, onde atualmente qualquer modelo básico permite medir o batimento cardíaco do utilizador, sendo que em alguns modelos mais avançados também já é possível realizar eletrocardiogramas de forma totalmente portátil e móvel [31].

Em casos mais específicos, como os pacientes diabéticos, existem também dispositivos com capacidade de monitorizar continuamente os seus níveis de glicose no sangue e informá-los atempadamente de uma possível desregulação dos valores. A ligação entre estes dispositivos e o *smartphone* do paciente permite que este mantenha um registo dos seus valores e que ajuste os seus hábitos de acordo com as necessidades do seu organismo [32].

Num contexto de *smart campus*, estes conceitos podem ser associados à monitorização de diversos parâmetros da comunidade estudantil que verifiquem o impacto que ambiente académico tem na sua saúde. É ainda possível relacionar a comodidade e a portabilidade destes dispositivos com outros dispositivos e tecnologias que permitam monitorizar a comunidade e identificar cada membro da mesma, como é o caso do uso de cartões RFID para acesso às instalações ou para registos de assiduidade.

A introdução de IoT na área do ambiente permitiu criar espaços mais saudáveis e de melhor qualidade para os seus utilizadores, promovendo em simultâneo, a sustentabilidade dos espaços e a economia de recursos. Seja este interior ou exterior, a monitorização de parâmetros que afetem a qualidade de utilização de um espaço, como a qualidade do ar ou da água [33], torna-se fundamental para garantir a segurança e o conforto dos seus utilizadores. A monitorização do consumo de recursos nesses espaços é igualmente importante, de forma a garantir a sustentabilidade dos mesmos, visto que o consumo ineficiente de recursos afetará inevitavelmente a qualidade e o equilíbrio ambiental desse espaço.

Os espaços inteligentes são geralmente dotados de diversos sensores e atuadores que permitem adaptar automaticamente o ambiente do espaço às necessidades e ao conforto dos utilizadores. Um exemplo de aplicação deste conceito é a ligação entre sensores de qualidade de ar, temperatura ou luminosidade e os respetivos atuadores como filtros de ar, sistemas de ventilação ou iluminação inteligente. Esta comunicação permite ajustar a intensidade ou até mesmo desligar os atuadores, para que os espaços possuam sempre as condições mais favoráveis aos utilizadores e para que os recursos não sejam desperdiçados [34].

Um sistema sensorial com capacidade de monitorizar todo o tipo de parâmetros ambientais que caracterizem a saúde ambiental dos espaços, podem ser denominados de sistemas de monitorização ambiental. Estes sistemas permitem alertar os utilizadores para possíveis emergências, como incêndios, cheias, poluição, entre outras situações [35].

Num contexto de *smart campus*, estes conceitos podem ser aplicados aos diferentes espaços interiores e exteriores que um *campus* possua, de forma a garantir o maior bem-estar e produtividade da comunidade académica. Os principais parâmetros a monitorizar e controlar num ambiente público e educativo são a qualidade do ar e da água, a iluminação, a temperatura e os níveis de ruído, visto que a análise em tempo real destes dados permite, por exemplo, ajustar automaticamente a climatização para evitar desconforto térmico ou a iluminação, de forma a evitar a fadiga ocular, permitindo ainda informar irregularidades na qualidade da água ou no nível de ruído sentido ao longo do *campus*.

2.1.2 Tecnologias de comunicação IoT

Tipicamente sem fios, as tecnologias de comunicação IoT são uma componente crucial de qualquer rede IoT, sendo que a escolha desta tecnologia irá determinar o seu desempenho, a sua sustentabilidade e o impacto que esta possui nos custos, nos consumos energéticos, na escalabilidade e na constituição da rede. As principais características que definem uma tecnologia de comunicação sem fios e a sua adequação aos requisitos específicos de uma implementação IoT são a capacidade de transmissão, a latência, o alcance, a segurança, os custos de implementação e o consumo energético. Dependendo do tipo de dados e da quantidade a ser transmitida, a capacidade de transmissão necessária pode implicar a escolha de uma tecnologia de comunicação que ofereça taxas de transmissão mais elevadas, como o 5G ou o WiFi, no entanto, se o consumo energético dos dispositivos for uma prioridade, esta escolha pode recair para tecnologias com menores consumos e conseqüentemente, menores capacidades de transmissão, como o Zigbee, BLE, NB-IoT ou LoRa [36]. Apesar de existirem outras, a escolha das tecnologias mencionadas para esta comparação deve-se à relevância que estas têm no mundo de IoT e ao facto de estas possuírem características diferenciadoras e abrangerem todo o tipo de aplicações IoT, servindo de base para comparação com qualquer outra tecnologia de comunicação.

O 5G trata-se da quinta geração de comunicações móveis e é uma das tecnologias de comunicação em maior crescimento em IoT devido à sua elevada capacidade de transmissão e muito baixa latência, conjugada com o fator móvel deste tipo de comunicações e alta densidade de cobertura, o que a torna ideal para aplicações IoT em tempo real e de atuação precisa, como os veículos autónomos ou cirurgias remotas e para aplicações no meio urbano e no contexto de cidades inteligentes ou *smart cities* [37], [38].

Através da utilização de frequências, na ordem dos 28 GHz, próximas da banda das ondas milimétricas (mmWave), esta tecnologia permite obter uma capacidade de transmissão de até 10 Gbps e latências inferiores a 1 milissegundo [39], [40]. O 5G possui ainda medidas de segurança 3GPP (*3rd Generation Partnership Project*), o que permite realizar comunicações seguras com capacidade para até 1 milhão de dispositivos por km² [41]. No entanto, quando utilizada a banda mmWave, o seu alcance é de apenas algumas centenas de metros e o seu custo de implementação é ainda bastante elevado, devido à necessidade de implementação de novas infraestruturas e maior densidade de antenas. No contexto de IoT, o seu consumo energético é também elevado, ainda que o 5G tenha introduzido uma redução deste consumo em relação às gerações anteriores de comunicações móveis [42], [43].

O WiFi é atualmente a tecnologia mais utilizada para aceder à Internet e para a criação de redes locais devido à sua simplicidade de implementação, atingida através do estudo e da utilização generalizada desta tecnologia em todo o mundo. Visto que atualmente é bastante comum encontrar redes WiFi em todo o tipo de infraestruturas e serviços, a implementação de dispositivos IoT com capacidade WiFi torna-se menos complexa e menos dispendiosa devido à existência prévia de infraestrutura [44]. O WiFi 6, também conhecido como 802.11ax, é geração mais recente da tecnologia WiFi e possui diversas melhorias em relação às gerações anteriores, nomeadamente na capacidade de transmissão, na eficiência energética e na sua escalabilidade [45]. Esta tecnologia é ideal para ambientes inteligentes e interiores de alta densidade de dispositivos complexos, como casas, escritórios ou fábricas, mantendo o seu alto desempenho através de técnicas inovadoras, como OFDMA (*Orthogonal Frequency-Division Multiple Access*) e MU-MIMO (*Multi-User, Multiple-Input, Multiple-Output*) [46]. Através da utilização de frequências na ordem dos 2.4 GHz, 5 GHz e 6GHz (WiFi 6E), esta tecnologia permite obter uma capacidade de transmissão de até 9.6 Gbps e latências típicas na ordem dos 10 milissegundos [47]. Devido à necessidade de utilização de novos *routers*, o seu custo de implementação é ainda considerado moderado, sendo que estes novos equipamentos integram novas medidas de segurança como o WPA3 (*Wi-Fi Protected Access*), um alcance mais alargado de aproximadamente 100 metros e um consumo energético médio comparativamente a outras tecnologias, embora seja possível adaptar este consumo às necessidades da rede [48].

O Zigbee destaca-se das restantes tecnologias, como o 5G ou o WiFi, pelo seu consumo energético muito reduzido e pela sua capacidade nativa de criação de redes *mesh*, o que lhe permite abranger um tipo de aplicações IoT totalmente diferente das restantes tecnologias. Uma rede *mesh* possibilita a transmissão de dados entre dispositivos até que estes cheguem ao destino pretendido, aumentando assim a robustez da rede, o número de vias de comunicação entre dispositivos e o alcance [49].

Esta tecnologia é amplamente utilizada na automação residencial e em *smart homes* por permitir criar uma rede totalmente interligada de dispositivos de baixo consumo de energia e manutenção, sem que o utilizador possua conhecimentos técnicos da tecnologia e sem a necessidade de contemplar os típicos desafios de alcance de comunicação entre os dispositivos e a *gateway* da rede [50]. Visto que se trata de uma tecnologia de baixo consumo de energia, é natural que o Zigbee possua uma capacidade de transmissão mais baixa de até 250 kbps, com uma latência de aproximadamente 100 milissegundos e um alcance máximo de aproximadamente 100 metros, sendo que estas características estão associadas à utilização mais típica da faixa de frequências de 2,4 GHz. Apesar da maioria dos dispositivos disponíveis no mercado utilizarem esta faixa de frequências, o Zigbee permite ainda utilizar a faixa de frequências de 868MHz (na Europa) para aplicações onde seja necessário um maior alcance em detrimento da capacidade de transmissão [51]. As características desta tecnologia, apesar de mais limitadas, são mais adequadas a um tipo de aplicação como a automação residencial do que as características do 5G ou até mesmo do WiFi, pois permitem uma implementação muito menos dispendiosa e complexa. O Zigbee possui ainda medidas de segurança AES-128 (*Advanced Encryption Standard*) e capacidade para centenas de milhares de dispositivos por rede, sendo que devido à sua capacidade *mesh*, a rede pode estender-se por diversos quilómetros, mantendo um custo de implementação baixo e um consumo energético igualmente baixo comparativamente a outras tecnologias [52].

O BLE foi igualmente desenvolvido com o objetivo de realizar comunicações de baixo consumo energético, servindo-se de uma tecnologia de comunicação largamente adotada e estudada como o *Bluetooth* e tendo sido incorporado como uma importante especificação a partir da versão 4.0 dessa mesma tecnologia. Esta especificação possui uma maior latência e um menor alcance, em comparação com uma utilização comum do *Bluetooth*, oferecendo, no entanto, uma redução significativa do consumo energético. Esta tecnologia é amplamente utilizada em *wearables* e *beacons* devido à sua facilidade de comunicação de dados e ao seu baixo consumo energético, permitindo que estes dispositivos tenham uma maior autonomia energética [53]. Visto que o BLE é uma especificação incorporada no *Bluetooth*, as suas características irão depender da versão de *Bluetooth* que está a ser utilizada, porém, na versão 5.0 desta tecnologia, o BLE possui uma capacidade de transmissão de até 2 Mbps, utilizando a faixa de frequências de 2,4 GHz e uma latência de até 10 milissegundos [54], [55]. Nos dispositivos mais comuns, como os *smartphones*, é possível ter apenas uma dezena de dispositivos ligados em simultâneo, enquanto que, no caso das *hubs*, geralmente utilizadas para comunicação com *beacons*, esse número pode chegar às centenas de dispositivos.

Esta tecnologia possui ainda um alcance de aproximadamente 100 metros, medidas de segurança AES-128, um custo de implementação baixo, devido à grande quantidade e diversidade de dispositivos equipados com *Bluetooth* atualmente e um consumo energético muito baixo comparativamente a outras tecnologias [55].

O NB-IoT ou Narrowband IoT, como o nome indica, é uma tecnologia de comunicação especialmente desenvolvida para IoT, com base nas funcionalidades das tecnologias de comunicação móvel e com o propósito de abranger áreas comparativamente maiores e com uma grande densidade de dispositivos, através de sinais com largura de banda reduzida. As suas funcionalidades são definidas pela 3GPP e pelas atualizações (*releases*) que realiza, sendo que esta foi pela primeira vez apresentada na “*Release 13*” de 2016, como uma tecnologia de comunicação rádio inovadora que podia coexistir com GSM (*Global System for Mobile Communications*), UMTS (*Universal Mobile Telecommunications System*) e LTE (*Long-Term Evolution*) na mesma implementação. As suas características são baseadas na tecnologia de comunicação LTE, à semelhança do BLE com o *Bluetooth* e no uso das suas faixas de frequência de 800 MHz, 900 MHz e 1800 MHz, no caso de uma implementação na Europa [56]. Devido ao seu grande alcance, esta tecnologia é amplamente utilizada em locais remotos ou com pouca infraestrutura de rede e é por isso, frequentemente associada a aplicações IoT de monitorização ambiental. Utilizando uma largura de banda de 180 kHz, esta tecnologia possui uma capacidade de transmissão máxima de 200 kbps, um alcance típico de 10 km, uma latência de até 10 segundos, medidas de segurança 3GPP e um custo de implementação moderado, devido ao elevado preço das *base stations* e um consumo energético baixo comparativamente a outras tecnologias [57], [58].

O LoRa é uma das tecnologias de comunicação mais relevantes na área de IoT que, como o nome indica, foi desenvolvida para comunicações de longa distância. Essa comunicação é realizada através da técnica de modulação de sinal *Chirp Spread Spectrum* (CSS) que utiliza um maior intervalo de espectro de frequência com base na utilização de sinais sinusoidais, cuja frequência varia linearmente no tempo (*chirps*), oferecendo uma maior robustez em comunicações de longo alcance com baixa potência de transmissão. O fator de dispersão característico desta modulação define a dispersão do sinal ao longo do tempo e permite ajustar a propagação às necessidades da transmissão de dados. Este tipo de modulação de sinal utiliza uma maior largura de banda, estando a sua operação limitada às bandas de frequência ISM (*Industrial, Scientific, and Medical*) atribuídas a cada região global (na Europa opera a 868 MHz, 2,4 GHz, entre outras). A utilização de larguras de banda mais elevadas oferece uma capacidade de transmissão de até 50 kbps, com uma latência entre 0,5 e 5 segundos e um alcance até 20 km quando utilizadas larguras de banda mais baixas e fatores de dispersão mais altos [62], [63], [64].

O LoRa oferece ainda medidas de segurança AES-128, com um custo geralmente moderado devido ao preço mais elevado das *gateways* contrabalançado com um custo mais reduzido dos dispositivos e um baixo consumo energético destes [65].

Comparativamente às restantes tecnologias mencionadas, o LoRa oferece algumas vantagens, com especial ênfase nas aplicações de longo alcance e redes de sensores que abrangem áreas de elevada dimensão. Relativamente ao 5G, apesar da sua elevada taxa de transmissão e baixa latência, o seu consumo energético elevado torna-se insuportável para o tipo de aplicações pretendidas e para cenários de baixo volume de dados transmitidos. Em comparação com o WiFi, o LoRa oferece menor segurança, mas a transmissão é muito mais eficaz em cenários de pior cobertura, o que em certas aplicações é crucial. Relativamente ao Zigbee e ao BLE e ainda que estas tecnologias consigam criar uma rede *mesh* de longo alcance, a robustez da comunicação fica mais reduzida a cada “salto” entre dispositivos, o que conseqüentemente, aumenta o consumo energético e os custos de implementação da rede. Em comparação com o NB-IoT, o LoRa possui menor consumo energético e custos de implementação mais reduzidos, mas também menor capacidade de transmissão, o que permite concluir que a escolha entre estas duas tecnologias dependerá dos requerimentos de capacidade de transmissão e de latência do cenário de implementação em questão [59].

Na Tabela 1 é possível observar a comparação entre as tecnologias de comunicação IoT mencionadas nesta secção, sendo que os valores apresentados são os que tipicamente se utilizam para descrever a potencialidade das tecnologias e para este tipo de comparação geral das características das mesmas. Através desta comparação é possível observar a diferenciação entre as tecnologias e obter uma visão geral da adequação de cada uma a conjuntos de cenários de implementação diferentes.

Tabela 1: Comparação entre tecnologias de comunicação IoT

	5G	WiFi 6	Zigbee	BLE	NB-IoT	LoRa
Transmissão	10 Gbps	9.6 Gbps	250 kbps	2 Mbps	200 kbps	50 kbps
Latência	< 1 ms	10 ms	100 ms	Até 10 ms	Até 10 s	0,5 a 5 s
Alcance	Centenas de metros	100 m	100 m	100 m	10 km	Até 20 km
Segurança	3GPP	WPA3	AES-128	AES-128	3GPP	AES-128
Custo	Muito Elevado	Moderado	Baixo	Baixo	Moderado	Moderado
Consumo	Médio a Elevado	Médio	Baixo	Muito baixo	Baixo	Baixo

Em resumo, apesar da sua enorme capacidade de transmissão, a característica diferenciadora do 5G é a sua baixa latência que lhe permite ser utilizado em cenários de atuação em tempo real de forma estável, sendo que em cenários menos críticos, onde também seja necessária uma alta capacidade de transmissão, o WiFi 6 é utilizado como uma alternativa mais económica em termos de custos e de consumos energéticos [43], [45]. As restantes tecnologias mencionadas fazem parte de um conjunto de tecnologias de comunicação IoT focadas no baixo consumo energético e na sua aplicação em dispositivos e sistemas mais limitados no seu poder de processamento e disponibilidade energética.

O Zigbee é principalmente utilizado na formação de redes de sensores e de ecossistemas interligados devido à sua capacidade nativa de comunicação *mesh* [49], enquanto o BLE é tipicamente utilizado na comunicação isolada entre dispositivos próximos e em aplicações que requerem ligeiramente menor latência e que possuem maiores volumes de dados [55]. Devido ao grande alcance de comunicação, o NB-IoT e o LoRa são geralmente utilizados para cobrir áreas de grandes dimensões e para comunicar pacotes reduzidos de dados a grandes distâncias, sendo que apesar de possuírem características e aplicações semelhantes, o LoRa é tipicamente mais utilizado devido à sua menor complexidade de implementação e maior alcance em zonas rurais [58].

As tecnologias de comunicação definem a camada física da comunicação, ou seja, a forma física da transmissão, receção e modulação de sinal, sendo que geralmente, diferentes tecnologias possuem diferentes métodos de comunicação de dados. De forma a uniformizar a comunicação entre dispositivos equipados com a mesma tecnologia de comunicação e a permitir a mesma entre dispositivos com tecnologias distintas, foram desenvolvidos protocolos de comunicação que definem regras e padrões de comunicação como o CoAP (*Constrained Application Protocol*) ou o MQTT [66].

O CoAP é um protocolo de comunicação desenvolvido para dispositivos e redes com recursos limitados, como as que se encontram em IoT. Este protocolo baseia-se no modelo REST (*Representational State Transfer*), utilizando métodos de comunicação *web*, como GET, POST, PUT e DELETE e opera ainda sobre o protocolo UDP (*User Datagram Protocol*), contribuindo para a eficiência e simplicidade das redes. As principais características deste protocolo incluem a possibilidade de realizar comunicação assíncrona, um cabeçalho reduzido de apenas 4 *bytes*, medidas de segurança DTLS (*Datagram Transport Layer Security*) e comunicação *unicast* mais eficiente [67].

O MQTT é um protocolo de comunicação simples e amplamente utilizado em redes IoT pela sua simplicidade e eficiência, especialmente na sua versão MQTT-S [68]. Este protocolo baseia-se numa arquitetura de comunicação de publicação e/ou subscrição de tópicos, no qual um dispositivo pode publicar mensagens agregadas a tópicos específicos (*publisher*) ou subscrever a esses tópicos para receber as mensagens (*subscriber*), sendo que os dispositivos comunicam sempre através de um servidor intermediário denominado “*broker*” [11]. O MQTT utiliza uma arquitetura de comunicação *broadcast*, o que significa que o *broker* comunica as mensagens associadas a um tópico X a todos os dispositivos que a este estiverem subscreitos, mas os dispositivos apenas comunicam com o *broker*. Na Figura 4 é possível observar um exemplo de comunicação MQTT, na qual a informação associada ao tópico “FAN” é transmitida de um dispositivo para outro, através do *broker*. Este protocolo suporta milhares de dispositivos ligados em simultâneo e possui ainda medidas de qualidade de serviço (QoS – *Quality of Service*) e de segurança (TLS – *Transport Layer Security*) que garantem a estabilidade da comunicação e a integridade dos dados [69].

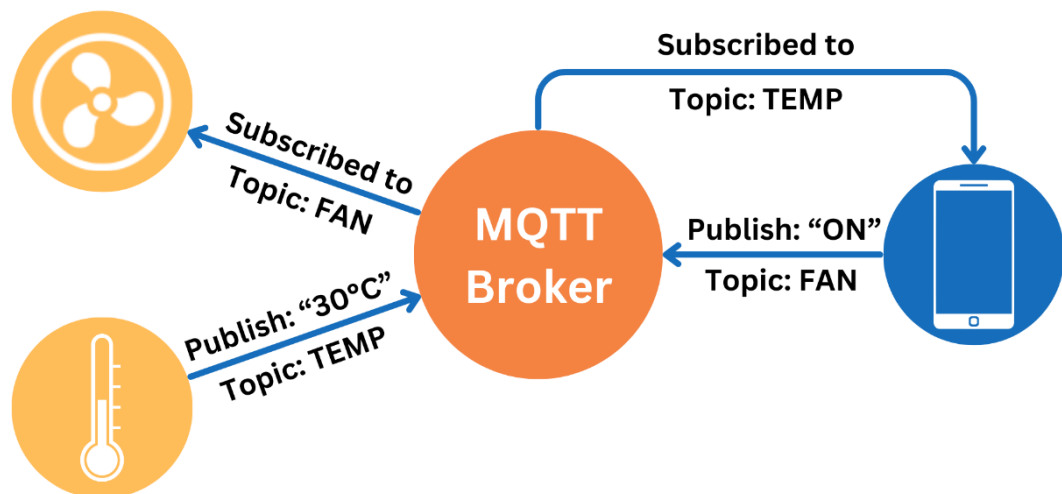


Figura 4: Exemplo de comunicação MQTT [11].

Visto que o CoAP e o MQTT são os protocolos mais utilizados em IoT, a Figura 5 apresenta uma comparação entre estes dois protocolos. A principal diferença entre estes refere-se à sua arquitetura de comunicação, sendo que o CoAP baseia-se numa comunicação mais divergente e descentralizada, enquanto o MQTT se baseia numa comunicação mais convergente e centralizada num intermediário.

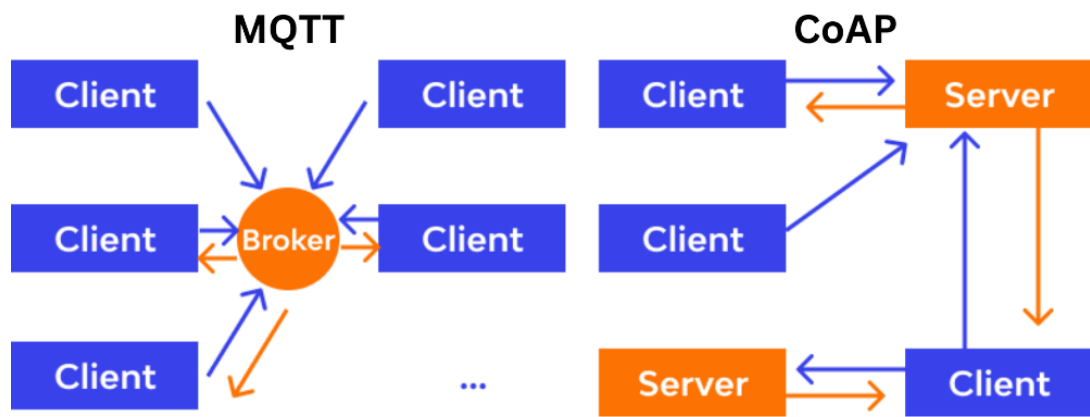


Figura 5: Comparação entre MQTT e CoAP [67].

2.2 LoRa

O LoRa é uma tecnologia de comunicação sem fios de longo alcance e baixo consumo energético, sendo uma das tecnologias mais importantes em IoT atualmente. A Semtech é a atual detentora da propriedade intelectual e das patentes relacionadas com o LoRa e tem sido a principal responsável pelo desenvolvimento e promoção da tecnologia, incluindo a sua comercialização e a integração em diferentes plataformas e aplicações IoT. Em 2015, a LoRa Alliance [76] foi fundada como uma associação sem fins lucrativos, constituída por diversas empresas e organizações dos ramos das telecomunicações e de IoT, de forma a padronizar a tecnologia através da criação de um protocolo de comunicação único e aberto denominado de LoRaWAN. O LoRaWAN está atualmente na sua versão 1.1, a qual foi publicada em 2017, sendo que a atualização mais recente foi realizada em 2020 à versão 1.0.4 [77].

O LoRa faz parte do conjunto de tecnologias de comunicação IoT focadas no baixo consumo energético e na comunicação entre dispositivos de recursos limitados, sendo que dentro desse conjunto, existe ainda um grupo de tecnologias com alcances de comunicação alargados que constituem uma LPWAN (*Low-Power Wide-Area Network*). Uma LPWAN representa uma extensa rede de dispositivos equipados com sensores e módulos de comunicação de baixo consumo energético e longo alcance de comunicação, sendo que as principais métricas que a definem são a eficiência energética, a escalabilidade e a cobertura. Esta denominação pode ainda ser utilizada para representar o tipo de tecnologias de comunicação utilizadas nestas redes como o LoRa ou o NB-IoT [78].

Em 2022, o mercado LoRa e LoRaWAN tinha um valor aproximado de 4,1 bilhões de US\$ e espera-se que até 2030, esse valor atinja os 48,4 bilhões de US\$, o que se traduz numa taxa de crescimento anual composta (CAGR) de 36,8%. Este crescimento da tecnologia LoRa deve-se também à crescente procura por soluções IoT mais acessíveis e voláteis ao nível da integração com outras soluções. O investimento em alternativas mais inteligentes, automatizadas e eficientes, como as *smart cities*, as infraestruturas inteligentes ou a monitorização ambiental, tem também contribuído para o crescimento do LoRa e para o desenvolvimento de novas aplicações em torno desta tecnologia de comunicação. Observando a Figura 6, é possível concluir que o setor industrial é aquele que maior quota de mercado possui atualmente, sendo que as previsões apontam para que em 2030, esta quota aumente e este setor de atividade represente 2/5 de todo o mercado [79].

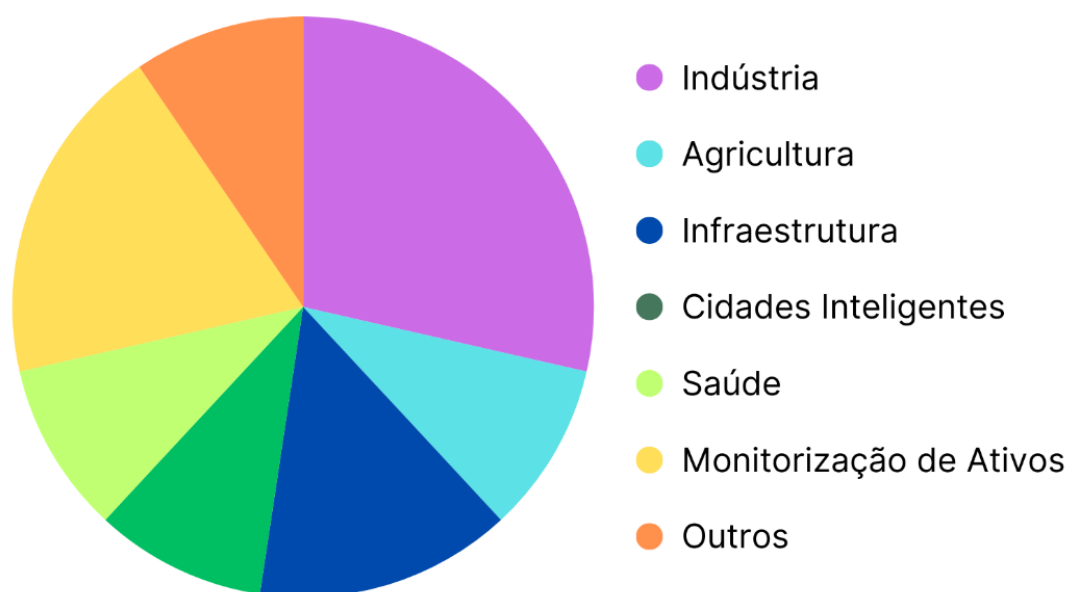


Figura 6: Quota de mercado LoRa e LoRaWAN por setor de atividade (2022) [79].

Os estudos realizados à tecnologia LoRa e ao protocolo LoRaWAN são especialmente importantes para o seu desenvolvimento devido à componente comunitária associada a esta tecnologia e são por isso, uma fonte de informação bastante abrangente de todas as potencialidades da mesma, seja ao nível das características físicas da comunicação ou ao nível de aplicações e implementações reais. O número de artigos científicos publicados sobre o LoRa tem crescido exponencialmente, devido à enorme potencialidade da tecnologia e à componente *open-source* do protocolo de comunicação, o que lhe confere diversas vantagens em relação a outras tecnologias. Estas publicações podem ser agrupadas em 5 categorias: Estudos, Avaliações Técnicas, Implementações Experimentais, Simulações e Aplicações [80].

Os estudos consistem na introdução ao LoRa e ao LoRaWAN e apresentam de forma geral as características e vantagens da tecnologia, servindo também como forma de apresentar as atualizações e melhorias realizadas. As avaliações técnicas consistem na análise dos diversos parâmetros que constituem o desempenho da tecnologia como a capacidade de transmissão, a latência, o alcance, a segurança ou o consumo energético. As implementações experimentais consistem no desenvolvimento e teste de protótipos em ambiente experimental, apresentado o funcionamento da tecnologia através de *hardware* e *software* de desenvolvimento. As simulações consistem na criação de redes LoRa através de *software* e permitem a configuração e simulação da rede, sem a necessidade de implementações físicas. Por fim, as aplicações consistem na apresentação de cenários de implementação da tecnologia e na reflexão da potencialidade da mesma [80].

Um dos objetivos das implementações IoT é a obtenção de um valor de custo/desempenho mais equilibrado, sendo que no contexto de LPWANs, isto traduz-se numa comunicação de longo alcance focada na redução do consumo energético e dos custos de implementação. Geralmente, as aplicações IoT que requerem a utilização do LoRa possuem duas características: comunicação sem fios em áreas de difícil acesso ou alta interferência e pacotes de dados de tamanho reduzido. A primeira característica associa-se ao LoRa devido às suas características de propagação e modulação de sinal, o que lhe permite transmitir dados a grandes distâncias e prevalecer sobre o ruído gerado por outros sinais de rádio ou obstáculos. O baixo volume de dados associado a estas aplicações contribui para que a baixa capacidade de transmissão do LoRa não seja um problema relevante na implementação e para que o consumo energético seja mais reduzido, visto que o volume de dados gerado por aplicações de sensorização pode tipicamente ser representado por simples estados (On/Off ou 1/0) ou valores de dimensão reduzida, como, por exemplo, temperatura ou humidade. O intervalo entre transmissões de dados é também tipicamente alargado e na ordem dos minutos, pelo que a latência das transmissões LoRa não constitui um obstáculo [81].

Uma rede LoRa é constituída por 4 elementos distintos: Nós Sensores, *Gateways*, Servidores de rede e Servidores aplicativos. Os nós sensores são responsáveis pela recolha e envio de dados relativos à sensorização e monitorização de diversos parâmetros, sendo que estes podem ainda possuir uma componente de atuação e automação de tarefas. Os dados recolhidos são enviados para uma *gateway* que intermedia esses dados para um servidor de rede, tipicamente através de outro tipo de tecnologias de comunicação mais comuns, como *Ethernet*, WiFi ou rede móvel. Além da recolha de dados, a *gateway* permite ainda enviar dados, comandos de atuação ou atualizações de *firmware* para os nós sensores [82].

O servidor de rede é responsável pela gestão dos dados e pela configuração geral da comunicação e dos parâmetros que permitem que esta ocorra de forma mais eficiente e segura. O servidor de rede serve ainda de intermediário entre a *gateway* e o servidor aplicacional, sendo que alguns destes servidores oferecem também a possibilidade de filtração e processamento prévio de dados, antes da sua comunicação para o servidor aplicacional. O servidor aplicacional é responsável pelo processamento, análise e apresentação dos dados, com o objetivo de retirar conclusões e obter informação a partir dos mesmos. A identificação de zonas de maior congestionamento ou de padrões mais eficientes de sinalização luminosa, a partir do registo de dados de tráfego, representa um exemplo de informação criada através da análise de dados. Tipicamente, estes servidores permitem criar uma interface de utilizador personalizada onde é possível visualizar a informação e atuar sobre a rede e os dispositivos, através do envio de dados para o servidor de rede que se encarrega de os comunicar aos nós sensores [82]. Na Figura 7 é possível observar uma representação geral de uma rede LoRa e dos elementos que a constituem.

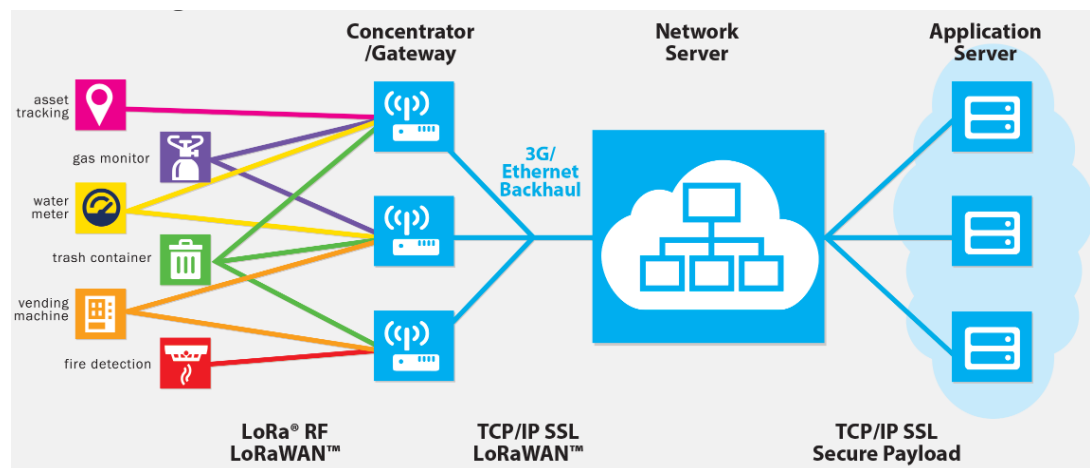


Figura 7: Representação geral de uma rede LoRa [82].

Uma rede LoRa divide-se ainda em 2 camadas: a camada física e a camada protocolar. A camada física é responsável pela emissão e receção dos sinais de rádio e por todos os fenómenos físicos relacionados com a tecnologia LoRa, nomeadamente, a modulação de sinal CSS, a banda de frequência e a intensidade do sinal, sendo que estes parâmetros são configuráveis e permitem ajustar a propagação do sinal às necessidades do utilizador e da rede [82]. Os *chirps* que constituem a modulação CSS podem ser denominados de *up-chirps*, caso a sua frequência aumente ao longo do tempo e *down-chirps*, caso a sua frequência ao longo do tempo diminua. Devido à linearidade da variação de frequência, a diferença entre a frequência recebida e a transmitida é encarada como uma variação de tempo e facilmente descodificada no recetor, o que permite reduzir a complexidade do processo de receção.

Sendo que pode existir uma variação de até 20% da frequência sem que existam impactos negativos na decodificação de informação, isto significa que não é necessária uma grande precisão na produção dos módulos de comunicação LoRa e que conseqüentemente o seu custo de produção se torna menor [59]. Na Figura 8 é possível observar um exemplo da modulação de sinal através de CSS e verificar o comportamento referido de variação de frequência.

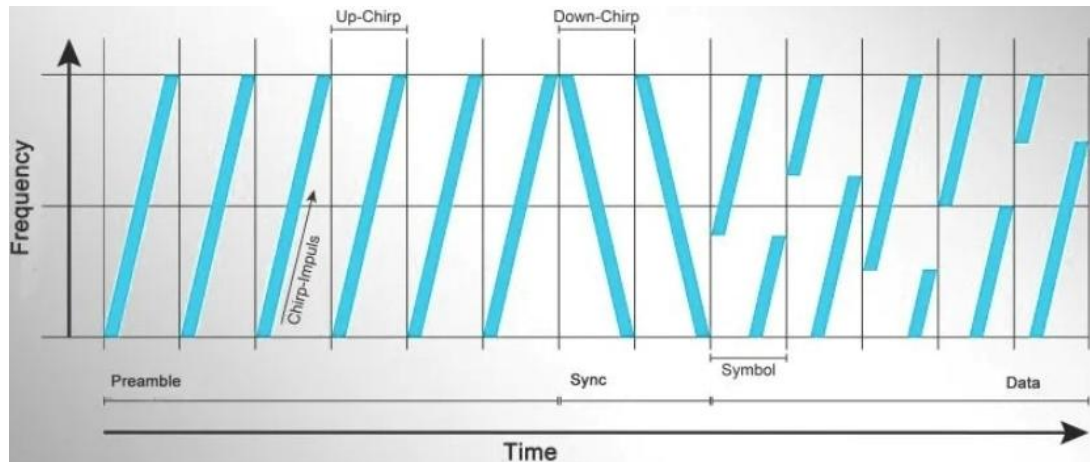


Figura 8: Exemplo de modulação CSS [60].

As características escolhidas para a modulação do sinal, nomeadamente, o fator de dispersão (SF – *Spreading Factor*), a largura de banda (BW – *Bandwidth*) e a taxa de codificação (CR – *Code Rate*) têm um impacto significativo no alcance, na capacidade de transmissão e no tamanho da mensagem que é transmitida. O SF caracteriza-se pelo número de *chirps* existentes em cada *bit* de informação transmitida e determina-se através do logaritmo de base 2 do número de *chirps* existentes em cada símbolo (Equação 1). Tipicamente, o SF varia entre 7 e 12, sendo que um $SF7$ oferece uma capacidade de transmissão maior e de menor alcance, enquanto um $SF12$ oferece maior alcance e robustez com uma menor capacidade de transmissão [59].

$$SF = \log_2 \left(\frac{\text{chirps}}{\text{símbolo}} \right) (=) \frac{\text{chirps}}{\text{símbolo}} = 2^{SF} \quad [1]$$

O tempo de transmissão de cada símbolo relaciona-se diretamente com o SF e com a BW utilizada, sendo que, quanto maior for o SF , mais “espalhado” estará o sinal ao longo do tempo e mais lenta será essa transmissão, apesar de possuir maior robustez e alcance. Em sentido contrário, quanto maior for a BW , menor será o tempo de transmissão, sendo que os valores tipicamente utilizados são 125, 250 e 500 kHz. A taxa de transmissão de símbolos é inversamente relacionada com tempo de transmissão, ou seja, neste caso quanto maior for a BW , maior será essa taxa. Através da equação 2, determina-se o tempo de símbolo e através da equação 3, determina-se a taxa de transmissão de símbolos [59].

$$T_{S[s]} = \frac{2^{SF}}{BW_{[Hz]}} \quad [2]$$

$$R_S \left[\frac{\text{símbolos}}{\text{segundo}} \right] = \frac{1}{T_S} = \frac{BW}{2^{SF}} \quad [3]$$

A tecnologia LoRa possui ainda um método de controlo de erros denominado de *Forward Error Correction* (FEC) que permite aumentar a robustez da transmissão e que se baseia no envio redundante de dados, para que o recetor identifique corretamente o conjunto de dados efetivamente enviado pelo emissor. Desta forma e como é possível observar na equação 5, para se determinar a capacidade de transmissão útil é necessário ter em conta a taxa de codificação CR e os “ SF ” bits de informação que são transmitidos em cada símbolo. A taxa de codificação varia entre 1 e 4 e é utilizada para determinar a taxa de correção de erro representada pela equação 4. A taxa de correção de erro é também por vezes representada por CR ou taxa de codificação, sendo importante fazer esta distinção e analisar a que valor é que se refere. Quanto maior for CR , maior será a proteção contra erros, o que conseqüentemente aumentará o tempo de transmissão e diminuirá a capacidade de transmissão útil [59].

$$\text{Taxa de correção de erro} = \frac{4}{4+CR} \quad [4]$$

$$R_b \left[\frac{\text{bits}}{\text{segundo}} \right] = SF * \frac{4}{2^{SF}} \quad [5]$$

De forma geral, um aumento da BW diminui a sensibilidade do recetor, enquanto que um aumento do SF aumenta a sensibilidade do mesmo. A análise da sensibilidade do recetor é importante para a avaliação do potencial da comunicação, uma vez que este estudo permite retirar conclusões sobre o alcance da mesma e sobre a adequação do módulo de comunicação ao local de implementação em questão. Uma das conclusões possíveis de retirar no caso do RSSI obtido ser muito superior à sensibilidade do recetor, é a existência de margem para que a distância ao emissor seja maior. No caso desse valor ser semelhante ao valor da sensibilidade, é possível que o módulo de comunicação utilizado ou a sua configuração, não sejam os mais adequados para aquele local de implementação, visto que se o sinal for recebido com menor intensidade, a comunicação não é possível. Estas conclusões relacionam-se com conceitos de *Link Budget*, RSSI e intensidade de sinal recebido, os quais serão aprofundados ao longo desta dissertação. Na Tabela 2, é possível observar o efeito das variações de BW e SF na sensibilidade do módulo de comunicação LoRa Semtech SX1276, sendo este um dos módulos LoRa mais utilizados no mercado. Na Figura 9 é possível observar a relação entre a modulação CSS e o fator de dispersão [59].

Tabela 2: Efeito da variação de BW e SF na sensibilidade do recetor [59]

Sensibilidade [dBm] – Semtech SX1276						
BW / SF	7	8	9	10	11	12
125 kHz	- 123	- 126	- 129	- 132	- 133	- 136
250 kHz	- 120	- 123	- 125	- 128	- 130	- 133
500 kHz	- 116	- 119	- 122	- 125	- 128	- 130

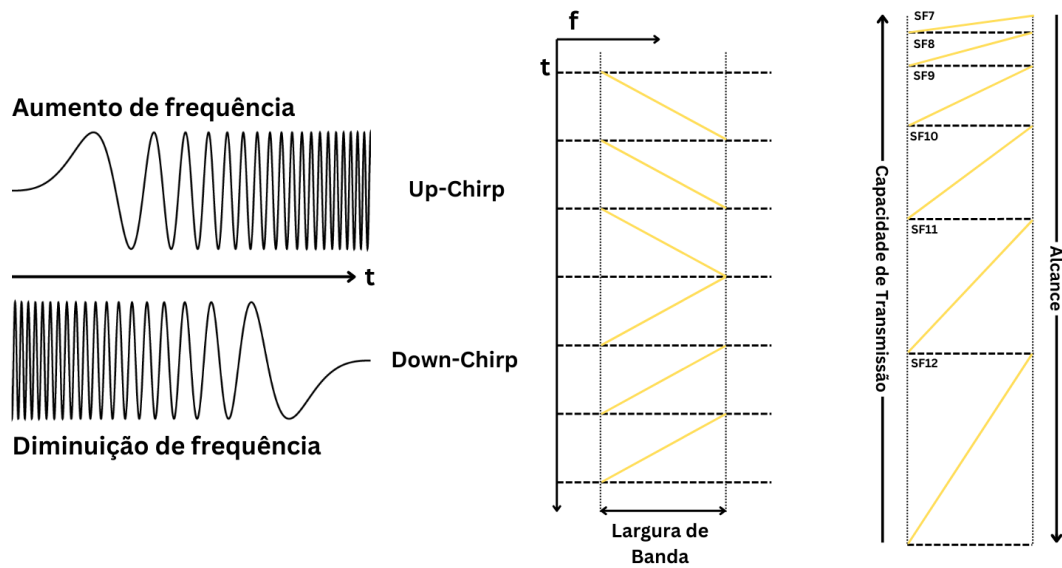


Figura 9: Relação entre a modulação CSS e o fator de dispersão [61].

A camada protocolar é responsável pela gestão da rede e pela organização dos dados, culminando no protocolo LoRaWAN referido anteriormente, o qual define a camada MAC (*Media Access Control*) da comunicação entre dispositivos LoRa e garante a segurança, a compatibilidade e a uniformização dos dados. Este protocolo permite ainda a atribuição de uma classe de comunicação a cada dispositivo, de acordo com as necessidades da implementação. Na Figura 10 é possível observar as camadas de comunicação da tecnologia LoRa [82].

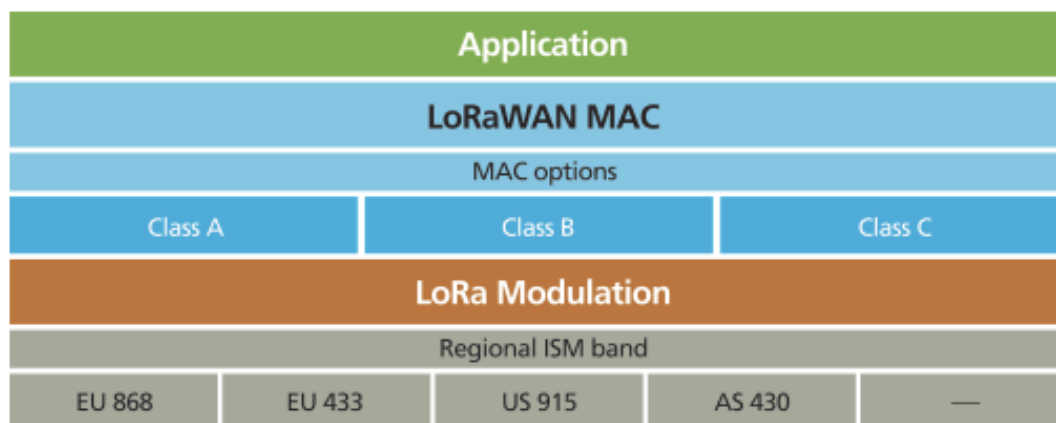


Figura 10: Camadas de comunicação da tecnologia LoRa [83].

A constituição de uma rede LoRa assemelha-se a uma rede de sensores, especialmente na constituição da rede física ou camada física, sendo que esta é constituída por nós sensores e pela *gateway*, dispostos numa topologia plana onde tipicamente prevalece a comunicação *unicast*. A agregação deste tipo de topologia com este tipo de comunicação constitui uma topologia em estrela, onde todos os dispositivos comunicam diretamente com a *gateway*. Esta organização da rede provém das próprias limitações da tecnologia, uma vez que os dispositivos não comunicam entre si como numa rede mesh [82]. Apesar de ser razoável afirmar que uma rede LoRa está limitada a esta arquitetura, é importante notar que existem diversos estudos a serem desenvolvidos de forma a tornar possível a comunicação entre dispositivos LoRa. Este tipo de comunicação entre dispositivos denomina-se de *Peer-to-Peer* [84].

O sinal modulado apresentado na Figura 8 pode ser representado através de uma trama LoRa, Figura 11. O sinal transmitido mantém constante a sua largura de banda e fator de dispersão ao longo de toda a transmissão e inicia-se por um preâmbulo responsável pela sincronização entre o emissor e o recetor. Este preâmbulo é composto por uma sequência constante de *up-chirps* que abrangem toda a largura de banda, sendo que os dois últimos *up-chirps* representam uma palavra de sincronização utilizada para diferenciar redes LoRa que utilizem a mesma banda de frequências e esta possui 1 *byte* de informação. O preâmbulo termina tipicamente com dois *down-chirps* que sinalizam o seu término e o começo do cabeçalho opcional [59]. Quando presente, este cabeçalho é transmitido com uma taxa de correção de erro de 4/8 e indica o tamanho do *payload* (em *bytes*), a taxa de codificação e a possível existência de uma verificação cíclica de redundância do cabeçalho, denominada de CRC (*Cyclic Redundancy Check*). A informação referente ao tamanho do *payload* é limitada ao tamanho de 1 *byte*, o que significa que o tamanho máximo do *payload* será de 255 *bytes* [59]. A constituição da trama LoRa termina com o *payload* e com a informação referente a uma eventual CRC do *payload* [59]. Na Figura 11 é possível observar a estrutura de uma trama LoRa.

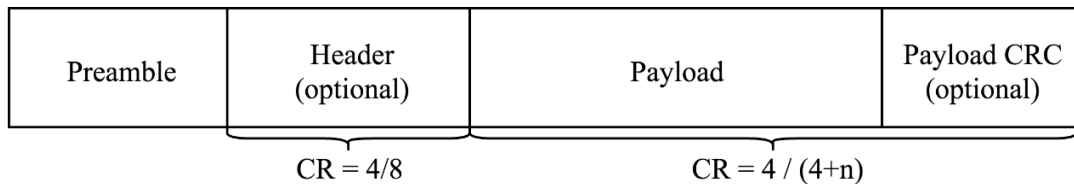


Figura 11: Estrutura de uma trama LoRa [59].

De forma a abranger as inúmeras aplicações IoT, os nós sensores LoRa carecem ainda de uma importante configuração ao nível protocolar LoRaWAN, que os permite adequar a sua comunicação ao cenário de implementação em questão. Esta configuração permite atribuir uma classe de comunicação A, B ou C, na qual a autonomia do dispositivo pode ser ajustada às necessidades de latência e de comunicação de dados. O método de comunicação destas classes baseia-se no método ALOHA, no qual os dispositivos transmitem dados de forma assíncrona e sem coordenação, ou seja, os dados são enviados sem que exista uma verificação prévia da disponibilidade do canal. Apesar deste método ter uma maior tendência para uma eventual colisão de mensagens, esta é mitigada pela reduzida taxa de transmissão de dados e pela utilização de técnicas como o CSS [85].

Por defeito, os dispositivos adicionados à rede são considerados de classe A, a não ser que sejam especificamente configurados para outra classe. Estes dispositivos possuem apenas duas janelas de receção de dados que só se abrem após uma emissão (*uplink*), ou seja, qualquer informação que seja enviada do servidor de rede (*downlink*), tem de aguardar por um *uplink* do dispositivo para que essa informação seja recebida nas janelas de receção do mesmo. Se o *downlink* ocorrer durante a primeira janela de receção, o canal a utilizar e as suas configurações serão as mesmas do *uplink*, caso o *downlink* ocorra na segunda janela de receção, utilizam-se configurações fixas e pré-definidas. Os dispositivos de classe A são os que consomem menos energia, por reunirem o seu consumo energético a um único momento de transmissão, no entanto, são os que oferecem menos flexibilidade de comunicação por existirem menos períodos de comunicação com a rede [85].

Os dispositivos de classe B possuem o rácio mais equilibrado entre autonomia de bateria e latência de comunicação devido a um maior número de períodos de receção. Para além dos períodos de receção presentes na classe A, possuem também períodos pré-definidos e agendados que permitem informar o seu estado de funcionamento ao servidor de rede e receber dados com intervalos periódicos bem definidos e de forma mais estável. Estes períodos são delimitados por *beacons* enviados pelo servidor de rede e permitem que o dispositivo se sincronize periodicamente com o servidor [85].

Os dispositivos de classe C possuem janelas de recepção praticamente contínuas que só fecham quando o dispositivo transmite informação. Este comportamento resulta num maior consumo de energia, mas oferece uma latência consideravelmente menor na recepção de informação [85]. Na Figura 12 é possível observar uma representação do comportamento das classes de comunicação do LoRaWAN.

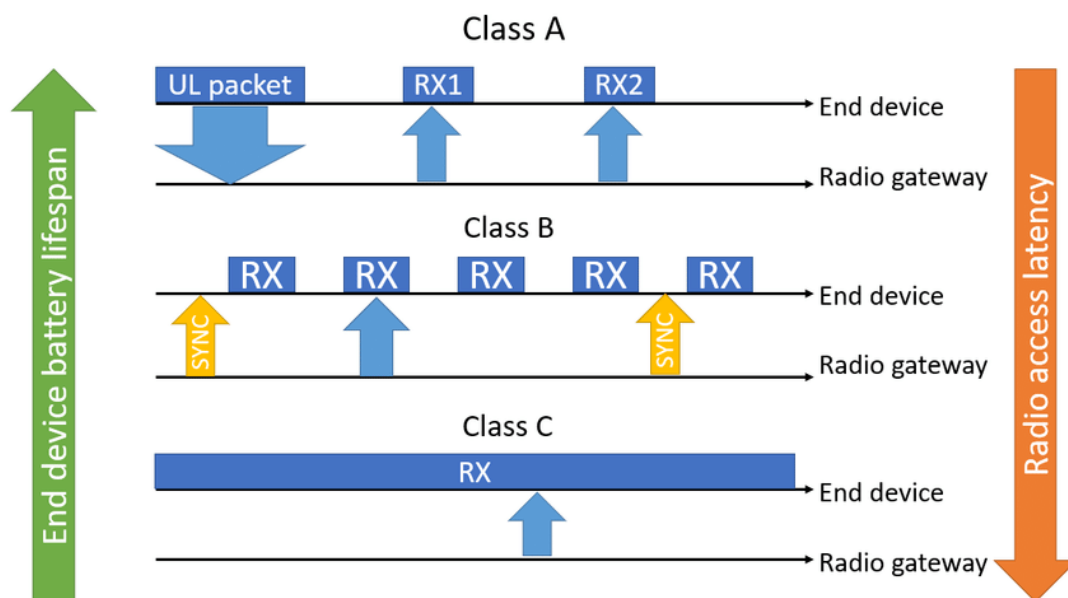


Figura 12: Comportamento das classes de comunicação do LoRaWAN [86].

O protocolo LoRaWAN possui métodos específicos de ativação de dispositivos na rede para que estes tenham chaves e identificadores únicos que os definem. Os dois métodos de ativação são OTAA (*Over-The-Air Activation*) e ABP (*Activation By Personalization*), sendo que ambos possuem características únicas relacionadas com a segurança, flexibilidade e gestão dos dispositivos. No método OTAA, os dispositivos conectam-se de forma dinâmica, solicitando um pedido *join-request*, ao qual o servidor responde com as chaves necessárias para a comunicação e com a informação *join-accept*. Estas chaves são geradas a cada novo processo de ativação, o que oferece maior segurança e flexibilidade, uma vez que estas chaves não precisam de ser estáticas e pré-programadas no dispositivo. Este método é ideal para redes dinâmicas compostas por diversos dispositivos onde a flexibilidade é fundamental, apesar de ser um processo mais complexo de implementar e no qual é mais comum existirem problemas de ativação [77].

No método ABP, os dispositivos conectam-se através de chaves pré-definidas e pré-programadas nos mesmos, o que torna o processo de ativação mais simples e sem necessidade de trocas iniciais de mensagens, apesar de o tornar também menos seguro e flexível. Na Tabela 4 é possível observar as chaves e identificadores necessários para cada método de ativação nas versões 1.0.x do LoRaWAN [77]. Devido às características da tecnologia LoRa, todos os dados transmitidos pelos nós sensores são recebidos pelas gateways que estiverem ao seu alcance e transmitidos aos servidores de rede associados.

No entanto, apesar destes servidores registarem essa ocorrência, estes só têm acesso à mensagem transmitida, caso possuam as chaves necessárias indicadas na Tabela 3.

Tabela 3: Chaves e IDs necessários para cada método de ativação [77]

Chave/ID	Descrição	OTAA	ABP	
Chaves necessárias antes da ativação				
AppKey	Utilizada para gerar a AppSKey e a NwkSKey e para calcular a integridade do <i>join-request</i>	Sim	Não	Pré-definida
Chaves necessárias depois da ativação				
AppSKey	Utilizada para encriptar o conteúdo dos pacotes de informação	Sim	Sim	Gerada através da AppKey e do <i>join-accept</i>
NwkSKey	Utilizada para encriptar os pacotes MAC e para calcular a integridade dos pacotes de informação	Sim	Sim	Gerada através da AppKey e do <i>join-accept</i>
Identificadores				
AppEUI	Identificador de 64 <i>bits</i> da aplicação do servidor de rede	Sim	Não	Pré-definida
DevEUI	Identificador de 64 <i>bits</i> do dispositivo	Sim	Não	Pré-definida
DevAddr	Identificador de 32 <i>bits</i> do dispositivo na rede	Sim	Sim	Gerada através do <i>join-accept</i>

Algumas das configurações do LoRaWAN, como a atribuição das classes de comunicação dos dispositivos e o processo de taxa de transmissão de dados variável (ADR – *Adaptive Data Rate*) são também geridos pelo servidor de rede. O processo ADR permite ajustar automaticamente a potência de transmissão e a taxa de transmissão de dados, através da alteração de valores de *SF* e *BW*, de forma a otimizar o consumo de energia e a qualidade da comunicação. Enquanto o servidor avalia a qualidade da comunicação através de parâmetros como a relação de sinal-ruído (SNR – *Signal-to-Noise Ratio*) e a intensidade de sinal recebido (RSSI), o dispositivo comunica com uma determinada taxa e potência de transmissão pré-definida pelo utilizador. Consoante a avaliação destes parâmetros, o servidor pode enviar comandos de configuração para o dispositivo que alterem os valores pré-definidos inicialmente. Para além destes parâmetros, existem ainda outros que são úteis à avaliação do desempenho da rede como a taxa de receção de pacotes, a percentagem de perda de pacotes, a taxa de sucesso de extração de informação ou o consumo energético geral da rede [59].

2.3 Link Budget e RSSI

O *Link Budget* é a denominação dada à determinação da potência de sinal recebida no recetor, através da conjugação de todos os ganhos e perdas da transmissão de sinal, sendo esta determinada através da equação 6. Através do *Link Budget*, é possível determinar teoricamente o alcance do sinal e analisar o resultado teórico da utilização de um determinado conjunto de valores de potência. Na Figura 13 é possível observar a representação das diversas componentes que compõem o *Link Budget* [64].

$$P_{Rx[dBm]} = P_{Tx[dBm]} + G_{Rx[dBi]} + G_{Tx[dBi]} - L_{Rx[dB]} - L_{Tx[dB]} - L_0[dB] \quad [6]$$

- P_{Rx} : Potência de sinal recebida no recetor.
- P_{Tx} : Potência de emissão.
- G_{Rx} : Ganho da antena de receção.
- G_{Tx} : Ganho da antena de emissão.
- L_{Rx} : Perdas na cablagem (receção).
- L_{Tx} : Perdas na cablagem (emissão).
- L_0 : Atenuação de propagação.
 - $L_0 = 10 * n * \log(d_{[m]}) + 20 \log(f_{[MHz]}) - 27,55$ [7]
 - n : Fator de atenuação.
 - d : Distância ao emissor.
 - f : Frequência de operação.

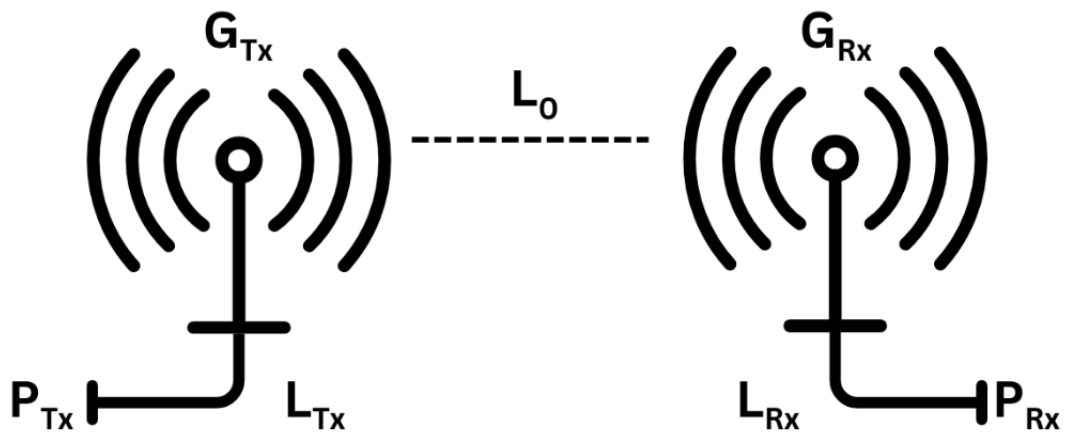


Figura 13: Componentes de *Link Budget* para transmissão de sinal.

A potência de sinal do emissor e os ganhos das antenas são valores tipicamente mais fáceis de obter, uma vez que estes podem ser configurados ou pré-definidos pelo fabricante do dispositivo. As perdas na cablagem referem-se à dissipação e à perda de potência que ocorre nos fios e nas ligações entre a antena e os dispositivos, sendo que estes valores são tipicamente mais complexos de determinar devido à quantidade de parâmetros físicos que impactam a qualidade e potência do sinal, como os contactos entre as ligações ou a largura e comprimento dos fios.

A atenuação de propagação refere-se à atenuação que ocorre durante a propagação do sinal no ar e depende do ambiente de propagação, da distância ao emissor e da frequência do mesmo, entre outros parâmetros. A atenuação de propagação é uma componente calculável e serve como uma referência teórica para determinar o alcance de comunicação, de acordo com o ambiente de propagação e com a frequência do sinal. A variável n no cálculo da perda de propagação de sinal varia consoante o ambiente de propagação, sendo que para um cenário de espaço livre, essa variável tem valor $n = 2$, enquanto para outros cenários, devido à presença de obstáculos e outros fatores, esse valor pode variar e representar uma perda de sinal mais ou menos acentuada devido a fenômenos como a reflexão, refração, difração ou absorção. Na Figura 14 é apresentado um exemplo de propagação de sinal, considerando o efeito de reflexão na superfície terrestre, o que a partir de determinada distância se traduz num valor $n = 4$.

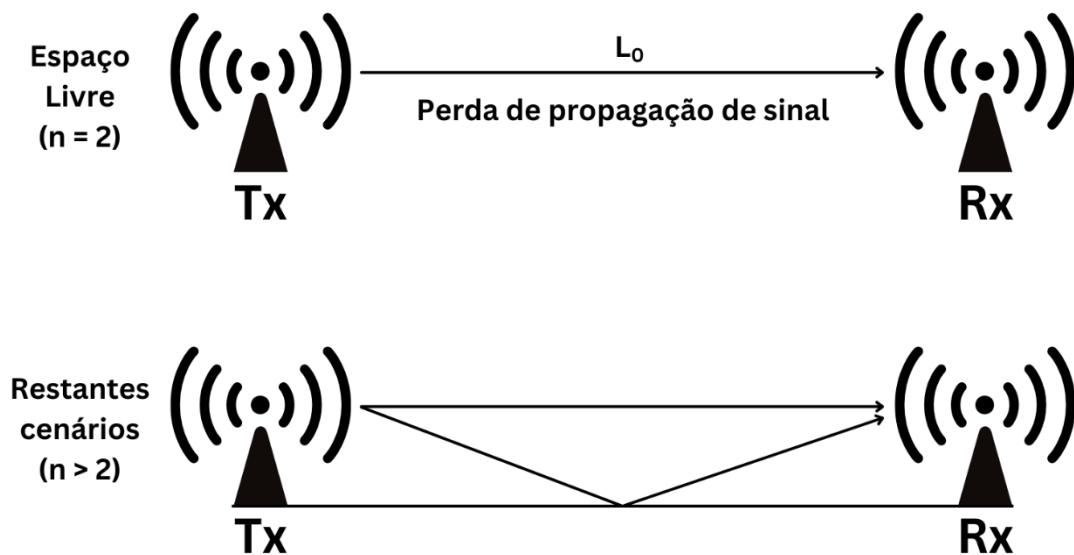


Figura 14: Influência da reflexão na superfície terrestre no parâmetro n .

Num planeamento de redes de comunicação sem fios, o estudo destes efeitos é de elevada importância para a robustez e para a eficácia de comunicação, nomeadamente, em sistemas de comunicação de longo alcance como o LoRa, onde a distância de comunicação e a dispersão de sinal são elevadas. Na prática, a atenuação do sinal pode ser maior do que a prevista teoricamente, uma vez que existem ainda outras perdas e outros fatores, como as condições atmosféricas que também contribuem para uma atenuação adicional do sinal. Para que a comunicação se realize corretamente, é então necessário que a potência de sinal recebida no recetor seja maior ou igual que a sensibilidade do mesmo, sendo que a sensibilidade do recetor representa o menor valor de potência com que este consegue realizar a comunicação. Em contexto prático, a potência de sinal recebida é obtida a partir do RSSI, o qual fornece um valor da intensidade do sinal recebida pelo recetor e é utilizado para monitorizar o desempenho da rede e aferir o modelo teórico de *Link Budget*, sempre que necessário [64].

3. Smart Campus

A qualidade dos serviços disponíveis à população têm um enorme impacto na comunidade onde estão inseridos, uma vez que esta qualidade define o grau de utilização dos serviços e o respetivo acesso à informação. A melhoria e a modernização destes serviços através da implementação de sistemas IoT tem-se tornado uma prática comum, nomeadamente, em serviços públicos e infraestruturas de grande dimensão, como os *campus* universitários. A IoT tem permitido desenvolver novos conceitos ligados à área do ensino superior, como os *smart campi*, que tanto contribuem para o processo de aprendizagem como para o ambiente e infraestruturas do *campus*. O conceito de *smart campus* tem sido aplicado de diversas formas em diferentes *campus* universitários, de acordo com as necessidades da comunidade estudantil e com os objetivos da administração da instituição de ensino. Neste capítulo será apresentado o conceito de *smart campus*, juntamente com aplicações específicas que o constituem, o impacto que este tipo de *campus* possui na qualidade da educação e dois casos de estudo que representam implementações reais de IoT em *campus* universitários.

3.1 Conceito de Smart Campus

O conceito de *smart campus* ou *campus* inteligente define um ambiente educativo que utiliza tecnologias como a IoT, com o objetivo de transformar o *campus* num espaço mais otimizado, eficiente e sustentável. Os *smart campi* conjugam espaços físicos com espaços digitais de forma a oferecerem uma maior acessibilidade a diversas informações e ferramentas obtidas através da implementação de sensores, atuadores e outras componentes IoT. Estes sistemas IoT permitem que a comunidade académica interaja e se envolva na melhoria dos serviços e das infraestruturas à sua disposição, com o objetivo de melhorar o bem-estar geral sentido no *campus* e o processo de aprendizagem [87].

Os *campus* podem ser vistos como pequenas cidades, visto que os serviços disponíveis à comunidade são semelhantes e que a principal diferença entre estes é tipicamente o tamanho e a escala do serviço. Por esse motivo, o conceito de *smart campus* e o conceito de *smart cities* partilham diversas semelhanças e aplicações em setores como a gestão de recursos, de pessoas ou de infraestruturas. Existem ainda outras aplicações IoT especialmente desenvolvidas para o setor da educação e que contribuem para um ensino adaptado a cada estudante. Um exemplo dessas aplicações são as salas de aulas inteligentes representadas por espaços educativos equipados com diversas tecnologias e dispositivos que contribuem para aulas mais eficientes e interativas, como sistemas de monitorização e controlo ambiental ou *softwares* educativos [87].

Em ambiente de aula, é importante garantir o bem-estar dos estudantes, mas é ainda mais importante obter um equilíbrio adequado entre a comodidade e a concentração na aprendizagem. Por esse motivo, as salas de aulas inteligentes possuem um ambiente controlado e monitorizado através de sensores de presença, iluminação, temperatura, humidade, entre outros, que permitem adaptar esse ambiente às melhores condições de aprendizagem. As presenças em aula podem também ser verificadas através de sistemas de leitura de cartões RFID ou de outros sistemas que maximizem o tempo útil de aula. Em detrimento de métodos educativos mais gerais e menos eficazes, a utilização de *softwares* educativos permite adaptar a aprendizagem ao tipo de estudante e ambiente da mesma, aumentando a oferta de educação e abrangendo novos tipos de estudantes, como trabalhadores-estudantes ou estudantes deslocados. Um dos métodos educativos em maior crescimento e que demonstra as vantagens da utilização de *software* educativo por parte de estudantes deslocados é a aprendizagem remota, a qual permite que os estudantes assistam às aulas e mantenham contacto com o ambiente educativo de forma remota, em qualquer lugar e a partir de qualquer dispositivo ligado à Internet [88]. Na Figura 15 é possível observar uma representação de diversas componentes e recursos IoT que constituem um *campus* inteligente.

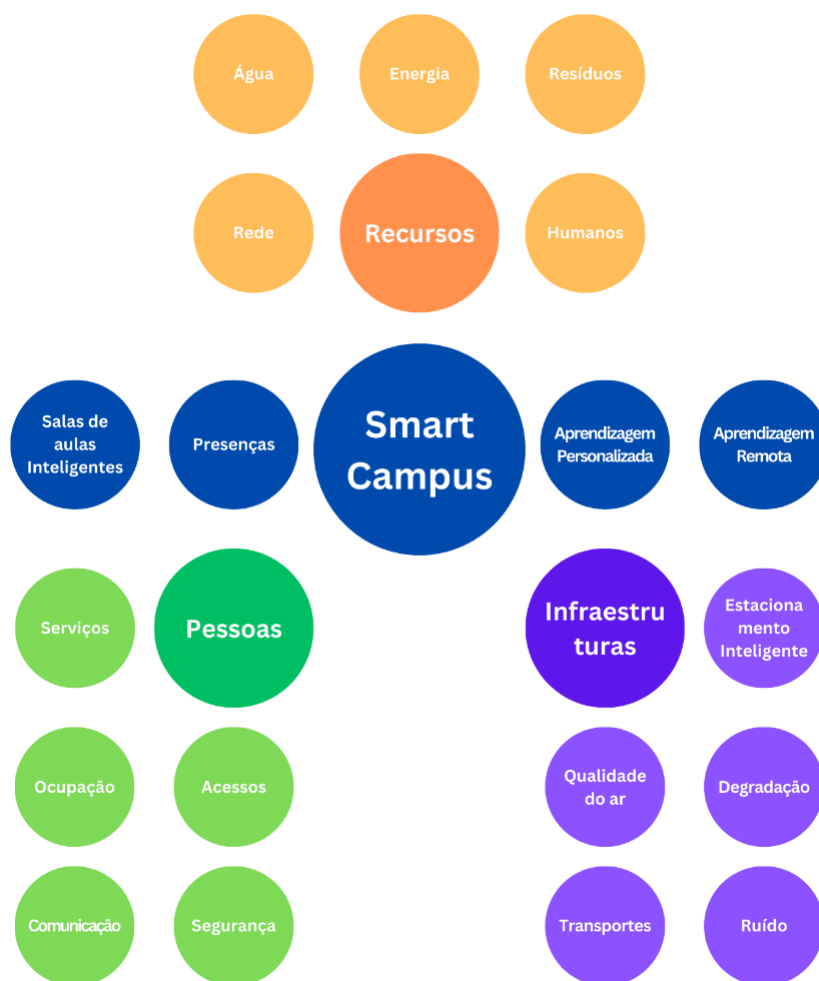


Figura 15: Componentes e Recursos IoT constituintes de um *campus* inteligente [88].

Os espaços abordados nos casos de estudo seguintes podem ser denominados de *smart spaces* e os ambientes que os rodeiam por *smart environments*, sendo que um conjunto de *smart spaces* constitui um *smart environment* e que um conjunto destes permite caracterizar um *campus* universitário como *smart campus*. Os *smart spaces* permitem identificar as necessidades dos seus utilizadores em tempo real e atuar sobre estas de forma personalizada, através dos sensores e atuadores presentes nestes espaços. Os dados obtidos permitem obter informação sobre a utilização dos espaços e adaptar as suas características ao bem-estar dos utilizadores, como a disposição da mobília, a temperatura ou luminosidade. Esta adaptação é executada conforme métodos estatísticos e de *machine learning*, o que contribui para uma intervenção humana mínima. Os *smart environments* são tipicamente espaços abertos ao público constituídos por espaços mais pequenos com tipos de utilização específicas e por esse motivo, constituídos por aplicações IoT específicas. Sendo que estes espaços agregam diferentes atividades e eventos, uma das suas principais características é a versatilidade e adaptação ao meio, já que o foco num único tipo de utilização não irá satisfazer todos os seus utilizadores. O uso de dispositivos pessoais do utilizador também é algo comum nestes espaços, o que acrescenta mais uma consideração a ter em conta nestas implementações, no que se relaciona com a compatibilidade dos dispositivos e poder de processamento dos mesmos. Nestas implementações é necessário ainda abordar a integração dos dispositivos numa só plataforma, a transparência e acessibilidade aos dados recolhidos e também a disposição e número de sensores/atuadores necessários à obtenção da maior eficiência e eficácia da implementação [90].

3.2 Casos de Estudo

A análise de casos de estudo sobre *smart campus* permite conhecer as diferentes tecnologias, abordagens e resultados obtidos em sistemas já implementados em ambientes educacionais, em particular, no ensino superior, sendo que a análise destes cenários é também fundamental para a identificação de lacunas e possíveis melhorias destes sistemas. Esta análise é constituída por duas implementações distintas de *smart campus*, tendo estas sido implementadas na Universidade de Oulu na Finlândia e na Universidade da Coruña em Espanha. De forma geral, os casos de estudo abordados englobam a implementação de diversos sensores que permitem avaliar os diferentes parâmetros físicos que definem a qualidade de utilização de um espaço, como temperatura, humidade, qualidade do ar, ruído, entre outros. Nos casos abordados foi utilizada a tecnologia de comunicação LoRa e foi demonstrado que a sua eficácia é bastante elevada para este tipo de aplicações, sendo que a sua utilização foi representada por diferentes dispositivos e servidores.

O primeiro caso de estudo analisado foi o da implementação IoT no *campus* de Linnanmaa da Universidade de Oulu na Finlândia. Esta implementação marcou o início de diversas iniciativas IoT neste *campus* e estabeleceu-se com a criação de um novo espaço com a denominação Tellus. O “Tellus Innovation Arena” é um espaço aberto desenhado para receber a comunidade estudantil, constituído por locais de estudo, trabalho colaborativo, apresentações, reuniões, descanso e ainda uma cafetaria. Através desta implementação IoT, pretende-se oferecer novos serviços e aplicações que atendam às necessidades educativas dos estudantes e avaliar o bem-estar dos utilizadores do espaço, de forma a aferir o impacto que este tem nos seus resultados académicos. Uma das principais questões em estudo relacionava-se com os efeitos da temperatura, da humidade e dos níveis de CO₂ no espaço, sendo que as variações destes valores não são lineares e dependem de diversos fatores como, estação do ano, hora do dia ou número de utilizadores. A exposição de luz natural no espaço possui um padrão mais estável, mas ainda assim se verificou diversas variações no padrão e utilizações ineficientes da iluminação, nomeadamente quando o espaço não estava a ser utilizado. A partir desta informação, concluiu-se que a utilização de padrões ou valores fixos torna-se uma prática ineficiente e que se torna necessário o desenvolvimento de algoritmos de adaptação às condições do ambiente. A implementação IoT realizada no Tellus consistiu na implementação de uma rede LoRa e de 331 nós sensores, distribuídos de acordo com a Figura 16 e equipados com duas pilhas de 3.6 Volts, GPS e sensores de temperatura, humidade, CO₂, movimento e luminosidade. Estes nós sensores comunicam a cada 15 minutos para uma *gateway* da Multitech ligada a uma antena biconica com ganho de 2 dBi. Os dados recolhidos são comunicados por MQTT à plataforma ThingWorx e são depois processados através de *scripts* Python, de forma a serem armazenados nos servidores locais do *campus* [90].

O servidor local está equipado com uma base de dados relacional PostgreSQL, um servidor Shiny que permite a criação de aplicações *web* através das linguagens R e Python e possui ainda uma *framework* Django REST que permite criar APIs para os dados recolhidos. Além dos nós sensores, este espaço está ainda equipado com 84 luminárias da Greenled controladas através do sistema ActiveAhead e das suas unidades de controlo, nas quais é possível encontrar individualmente um sensor de movimento e de luminosidade, que permite desligar a luminária autonomamente quando ninguém está presente ou ajustar a sua luminosidade de acordo com a luz natural do espaço, de forma a diminuir o consumo energético. As unidades de controlo comunicam entre si por BLE e possibilitam a criação de uma rede *mesh* com ligação à rede WiFi do espaço, a partir da qual é possível controlar as luminárias [90].

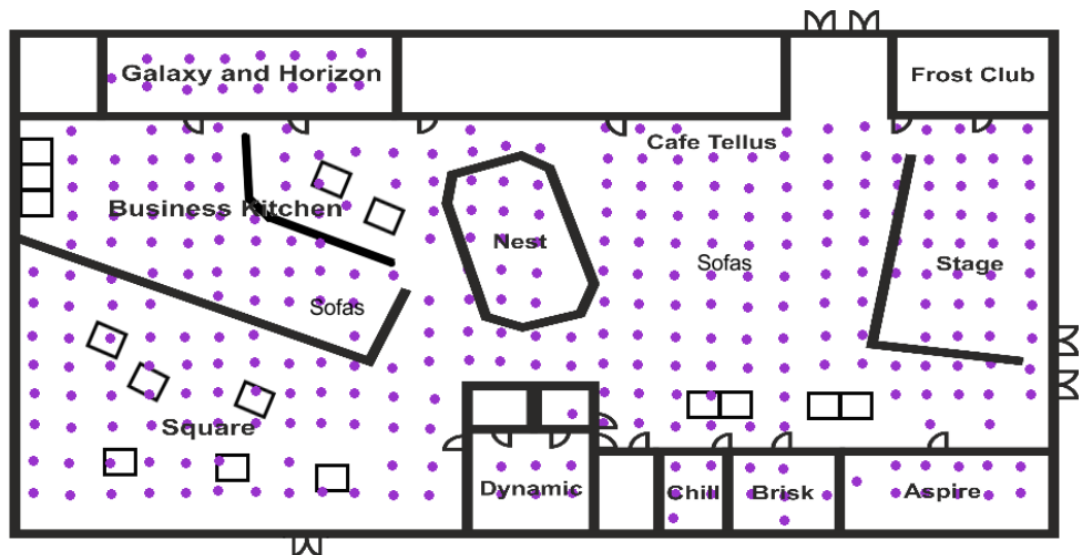


Figura 16: Distribuição dos nós sensores no espaço Tellus [90].

A visualização de dados recolhidos neste espaço é efetuada através de uma *dashboard* em formato de aplicação *web* ou através de uma aplicação de realidade aumentada (AR). A Figura 17 representa a página inicial da *dashboard*, na qual é possível observar um mapa do estado atual do ambiente, através da coloração de zonas de acordo com os valores do parâmetro escolhido (*heatmap*). Esta *dashboard* permite a criação de mecanismos de notificação e a realização de diversas análises e criação de gráficos individuais relacionados com os parâmetros recolhidos e com a localização pretendida dentro do espaço. Esta visualização e análise de dados é conseguida através de *scripts* R e utilizando a ferramenta Shiny, de forma a oferecer uma visão mais abrangente do espaço e dos padrões de comportamento que dentro dele ocorrem [90].

Tellus state for last 15 minutes

● Temperature ○ CO2 ○ Humidity ○ Light ○ Motion

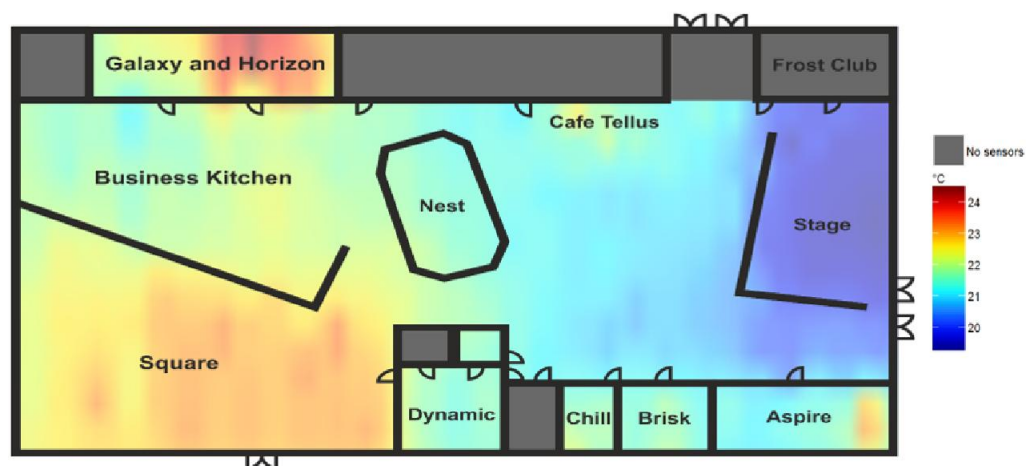


Figura 17: Página inicial da *dashboard* Tellus [90].

Além da *dashboard*, foi também desenvolvida uma aplicação que permite visualizar os dados de forma interativa, recorrendo à realidade aumentada. A primeira versão desta aplicação permitia observar os valores de temperatura individualmente recolhidos por cada sensor, ao direcionar a câmara do dispositivo móvel para o nó sensor. Esta aplicação tinha o propósito de assistir a equipa de manutenção na localização de nós sensores avariados. A segunda versão consistia na integração de objetos digitais no espaço que alteram as suas formas ou cores consoante as condições ambientais, sendo que nesta versão foram implementadas plantas para descrever as condições de temperatura e humidade do espaço. Na Figura 18 é possível observar a aplicação móvel desenvolvida para Android através da ferramenta Unity e do *plugin* ALVAR para funcionalidades AR [90].



Figura 18: Integração de objetos digitais interativos no espaço Tellus [90].

Neste tipo de espaços públicos, a adaptação do ambiente pode ser estudada e automatizada de acordo com as variações dos parâmetros ambientais, mas ainda assim, cada indivíduo tem as suas preferências. Desta forma, os investigadores responsáveis pela implementação IoT no espaço Tellus decidiram criar um sistema interativo para as luminárias que estivesse de acordo com a preferência dos utilizadores presentes. A aplicação móvel desenvolvida permite ajustar a luminosidade das luminárias próximas ao utilizador, através de uma abordagem que tem em consideração a preferência e a proximidade às mesmas, ou seja, a preferência do utilizador que estiver mais próximo da luminária vai pesar mais no valor de luminosidade, do que a preferência de outro utilizador mais afastado. O sistema utiliza uma interface NFC para iniciar o controlo das luminárias, no qual o utilizador é guiado para a aplicação móvel e instruído a ligar-se ao WiFi do espaço, para que a interface de atuação se apresente no dispositivo. A preferência do utilizador é enviada para o servidor, onde esta é conjugada com os restantes dados presentes e onde é calculado o valor de luminosidade que será enviado para as luminárias. Quando um utilizador se desconecta da rede WiFi ou da aplicação móvel, os níveis de luminosidade são recalculados, para que exista uma maior economia de energia e a luminosidade esteja sempre adaptada em tempo real às preferências dos utilizadores. A Figura 19 demonstra um exemplo do resultado da abordagem utilizada, no qual a luminária apresentada a verde é partilhada por dois utilizadores e a Figura 20 demonstra o aspeto da aplicação móvel utilizada para as controlar [90].

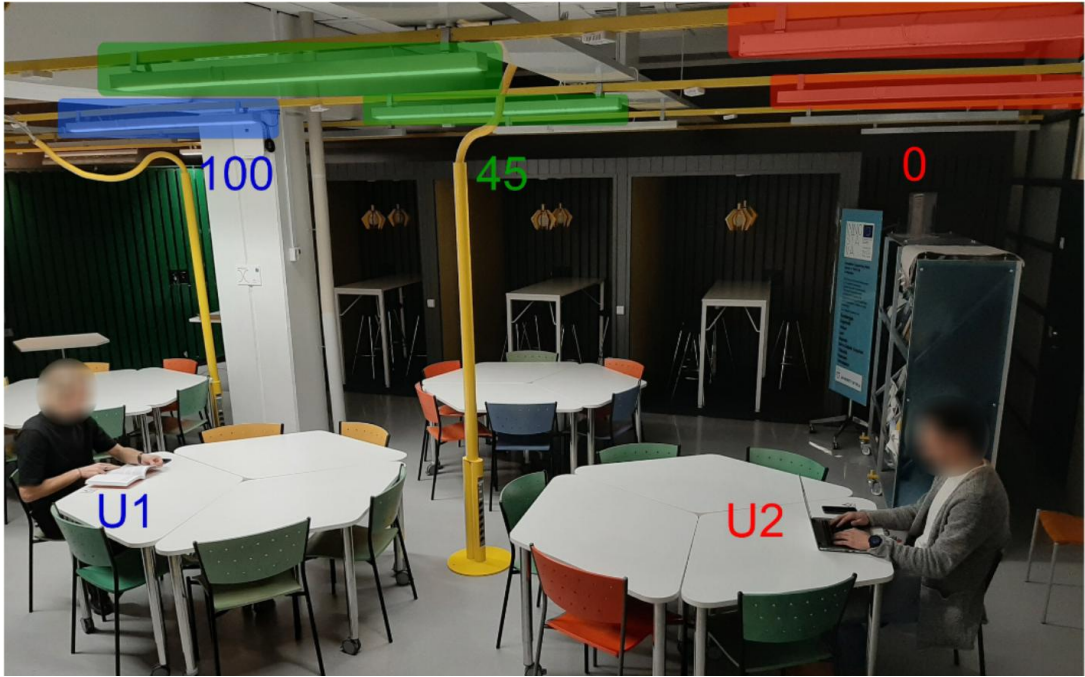


Figura 19: Resultado da abordagem utilizada nas luminárias [90].

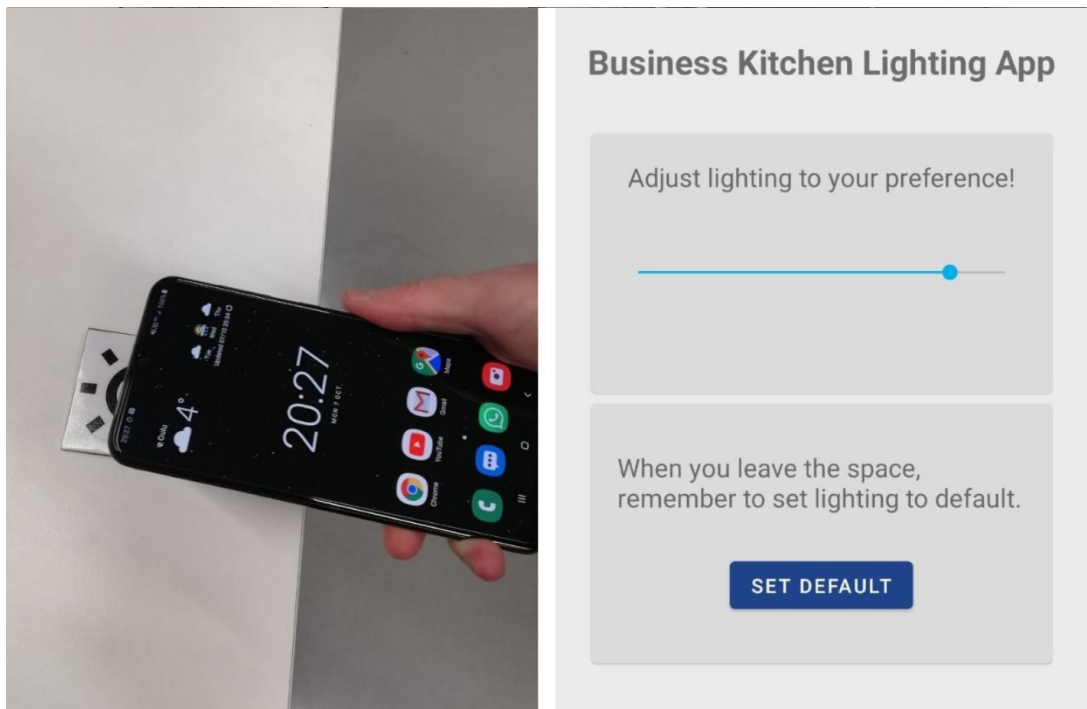


Figura 20: Aplicação móvel de controlo das luminárias [90].

A implementação da rede IoT e os dados recolhidos neste espaço contribuíram para o desenvolvimento de diversos projetos e estudos conduzidos pela comunidade académica e para a criação de pelo menos três cursos relacionados com IoT e com aplicações resultantes da rede. Estes resultados demonstram o impacto positivo e as possibilidades educativas que uma implementação IoT e a construção de um *smart campus* oferecem a toda a comunidade académica [90].

Após uma análise de utilização e um estudo mais aprofundado à rede implementada, os responsáveis deste caso de estudo apresentam ainda as melhorias que irão introduzir na rede e algumas considerações relativas à sustentabilidade da mesma. Uma das melhorias a introduzir é a instalação de um sistema autónomo de ar condicionado que se adapte automaticamente e forneça as melhores condições durante as horas de pico de utilização do espaço, nas quais os níveis de CO₂ se tornam particularmente desconfortáveis. A introdução de sistemas de humedificação do ar torna-se também necessária para a regulação da humidade, uma vez que a variação de temperatura natural da geografia de Oulu, origina extremos de humidade em diferentes alturas do ano. Nos nós sensores implementados podem ainda ser integrados sensores que permitam identificar fontes de ruído possivelmente perturbadoras para os utilizadores do espaço. Nos testes efetuados à aplicação móvel, os responsáveis identificaram ainda outros objetos digitais que seriam benéficos para representar os dados recolhidos em tempo real, como linhas de circulação de ar ou esferas flutuantes para representar os níveis de CO₂ [90].

Relativamente à sustentabilidade e manutenção da rede, os responsáveis apontaram algumas considerações importantes que poderão ser comuns a outros casos de estudo. A redundância de dispositivos é um método que contribui para um funcionamento contínuo e preciso da rede, sendo que este garante que os valores continuam a ser recolhidos nas imediações daquele local e permite verificar se algum sensor está a recolher dados de forma incorreta. Apesar de o custo ter sido maior, a implementação de 331 nós sensores no espaço Tellus garante que toda a sua área está coberta pela rede e pela sensorização do sistema [90]. O tratamento de dados para detetar incoerências e dados redundantes é também bastante importante para a eficiência da rede, pois permite que as análises de dados retirem conclusões mais acertadas e que a base de dados contenha apenas dados pertinentes, sendo que este tratamento tem de estar adaptado a todo o tipo de dados que possam ser recolhidos pelos nós sensores. A diversidade de sensores e de dispositivos numa rede IoT pode ter as suas vantagens, mas também implica uma maior complexidade e um maior número de diferentes integrações com a rede, o que pode gerar falhas na comunicação ou dificuldades no tratamento de dados. Sendo que a manutenção da rede é tipicamente assegurada pela equipa responsável pela sua implementação, o número de pessoas com acesso, conhecimento e experiência com a rede tende a ser limitado, o que pode originar a perda de conhecimento ao longo do tempo e uma manutenção mais pobre.

A segurança da rede é também uma questão bastante importante, principalmente quando a rede tem atuadores integrados e quando esses podem ser manipulados de forma remota. A segurança na transmissão e no armazenamento dos dados garante que estes não são manipulados de forma maliciosa e que se mantêm coerentes quando comunicados entre os diferentes sistemas integrados [90].

O segundo caso de estudo analisado foi o da implementação IoT na Universidade da Coruña em Espanha. Este caso de estudo baseou-se na criação de um sistema IoT onde fosse possível reunir, monitorizar e controlar dispositivos de diferentes fabricantes numa só plataforma IoT. A principal motivação para a necessidade de agregação dos dispositivos numa só plataforma proveio dos efeitos causados pela pandemia de COVID-19 e da dificuldade que a universidade teve em se adaptar rapidamente às novas condições ambientais e de saúde pública. De forma a precaver-se de situações semelhantes, a universidade decidiu monitorizar de forma mais eficaz e eficiente, a qualidade do ar e os níveis de CO₂, através de sensores previamente adquiridos e que até então estariam associados a uma plataforma proprietária do fabricante. Tanto estes sensores como outros implementados ao longo da universidade, possuíam plataformas e tecnologias de comunicação próprias fornecidas pelo fabricante, o que aumentava a dispersão e a complexidade da recolha de dados dos muitos sensores existentes. Esta situação motivou a criação de uma plataforma única que permitisse integrar todos os sensores e toda a informação recolhida, para que esta pudesse ser analisada de forma conjunta e mais eficiente [91].

Além dos sensores referidos, foram implementados outros sensores num dos centros de investigação da universidade, de forma a monitorizar diversos parâmetros físicos do ambiente, como a temperatura, humidade e níveis de CO₂ e ainda, no departamento responsável pela gestão de infraestruturas da universidade, foram implementados sistemas de monitorização de produção e consumo de energia e sensores mais específicos, como um sensor de concentração de gás rádon. No caso de estudo da Universidade de Oulu foram identificados diversos pontos de consideração, aos quais esta implementação deu resposta, como a interoperabilidade, custos, dados, disponibilidade, escalabilidade, manutenção e segurança, os quais se encontram retratados nos pontos seguintes [91].

- Interoperabilidade: De forma a ultrapassar os obstáculos inerentes à utilização de sensores de diferentes fabricantes, é necessária a implementação de uma plataforma integradora que tenha a capacidade de receber dados em diferentes formatos e de diversas fontes. A Home Assistant foi a plataforma escolhida pelos responsáveis desta implementação devido ao enorme catálogo de integrações que esta possui, o que permite interligar imensos dispositivos de diferentes fabricantes e tecnologias e ainda devido à enorme comunidade que a suporta.

- Custos: A implementação de uma rede IoT pode ser dispendiosa por envolver infraestrutura especializada e um grande número de dispositivos, o que implica uma avaliação aprofundada, tanto aos custos iniciais de implementação como aos custos de manutenção. De forma a reduzir os custos, a universidade decidiu implementar a rede de forma local e num dos seus servidores já existentes, utilizando ferramentas gratuitas e *open-source*.
- Dados: Tipicamente, uma rede IoT produz um enorme volume de dados proveniente do grande número de dispositivos que a compõe, o que implica um complexo processo de armazenamento, processamento, análise e acessibilidade. O sistema IoT deve ser capaz de realizar esse processo de forma eficiente e posteriormente apresentar os dados de forma interativa e de fácil compreensão. Sendo que a plataforma possui uma integração direta com a InfluxDB, esta foi utilizada como base de dados para armazenar o grande volume de dados a longo prazo. Para a criação da interface de visualização de dados, foi utilizada a Grafana que facilmente consegue recolher dados da InfluxDB através de *queries*.
- Disponibilidade: A disponibilidade do sistema e o seu funcionamento contínuo é algo fundamental em aplicações IoT empresariais, sendo que é tipicamente exigido um valor mínimo de *uptime*, às empresas detentoras de ferramentas IoT na *cloud*. Visto que o sistema a implementar não é crítico e que o seu acesso é limitado à comunidade académica, a disponibilidade não é um ponto crucial nas considerações, mas ainda assim, foi tida em consideração a implementação do sistema em máquinas virtuais instaladas em servidores modernos e com baixa probabilidade de falhar.
- Escalabilidade: Para que o sistema seja escalável, é necessário que a rede tenha capacidade e espaço para a introdução de mais dispositivos e para um maior volume de dados. De forma a aumentar os recursos da rede é possível escalar os sistemas de forma vertical, no qual se adicionam mais recursos aos sistemas atuais ou de forma horizontal, no qual se acrescentam mais sistemas com poder de processamento semelhante. A escala vertical dos sistemas é tipicamente menos complexa e mais compatível com as ferramentas existentes, sendo que as versões *open-source* das ferramentas utilizadas nem suportam um escalar horizontal dos sistemas. Caso a rede implementada se torne saturada, a universidade planeia realizar um escalar vertical, sendo que esta é também a opção mais adequada para redes IoT de menor dimensão.

- **Manutenção:** A manutenção e monitorização de redes IoT é especialmente importante para garantir um funcionamento eficaz, eficiente e contínuo, nomeadamente, em redes IoT sem fios, onde as falhas de comunicação são comuns. Esta manutenção envolve a verificação periódica da precisão do dispositivo, a atualização de *firmware*, a troca ou carregamento de baterias, entre outras operações que garantam um bom funcionamento do dispositivo e dos sistemas. Através da interface do Grafana, é possível observar os dados recolhidos pelos nós sensores e as informações relativas à bateria e à comunicação do mesmo. Como forma de monitorização e alerta para a manutenção dos dispositivos, é ainda utilizado o Node-RED, no qual foi criada uma automatização de envio de notificações, de acordo com os níveis de bateria dos dispositivos.
- **Segurança:** Os dispositivos IoT implementados recolhem dados sobre diferentes parâmetros físicos que envolvem a comunidade e o ambiente de implementação, o que significa que existe alguma sensibilidade nestes dados e é necessário que estes sejam comunicados com a máxima segurança. De forma a prevenir o acesso e adulteramento dos dados, é comum utilizar-se protocolos de encriptação de dados, como o HTTPS, sendo este o protocolo utilizado nesta implementação IoT para todos os serviços e dispositivos diretamente expostos à Internet. Visto que alguns dispositivos só funcionam com comunicação desenscriptada, como é o caso do BACnet ou do Modbus, estes são ligados a uma VLAN criada especificamente para este tipo de comunicações.

A implementação IoT realizada na Universidade de Coruña consistiu na implementação de uma rede LoRa e de 52 nós sensores Aranet4 Pro, equipados com sensores de temperatura, humidade relativa, CO₂ e pressão atmosférica. Estes nós sensores conseguem comunicar em intervalos de 1, 2, 4 ou 10 minutos para uma *gateway* do fabricante, através de LoRa e com um alcance máximo de 3 km. Na Figura 21 é possível observar a arquitetura de toda a implementação e verificar que foram utilizadas 3 *gateways* Aranet Pro, que comunicam os dados recolhidos por MQTT à plataforma Home Assistant.

Os dados recebidos de cada nó sensor representam seis parâmetros: CO₂ [ppm], temperatura [°C], humidade relativa [%], pressão atmosférica [hPa], bateria [%] e RSSI [dBm]. Esta implementação culminou com o teste em ambiente real dos exames de entrada na universidade em Espanha, onde se espera um grande número concentrado de pessoas nas salas de aulas. A colocação de uma *gateway* por edifício foi suficiente para abranger os diversos nós sensores instalados em cada um deles e os dados obtidos, representados na Figura 22, demonstram as variações dos níveis de CO₂ ao longo do primeiro dia em que os exames se realizaram. Estes dados permitem determinar a qualidade do ar nos edifícios e o grau de necessidade e utilização de sistemas de ventilação e purificação do ar, para que as condições sejam as mais favoráveis à comunidade estudantil [91].

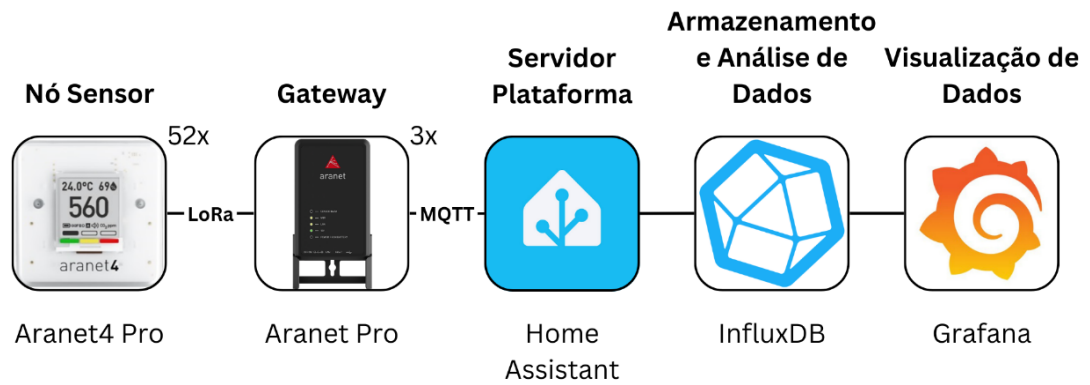


Figura 21: Arquitetura IoT da Universidade de Coruña [91].

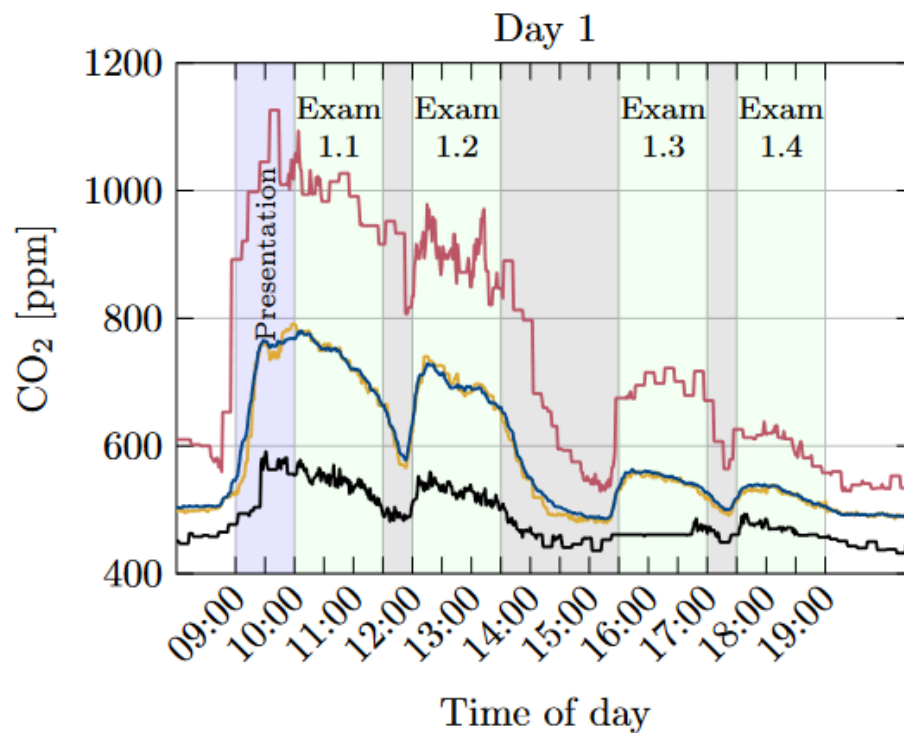


Figura 22: Variações dos níveis de CO₂ ao longo do primeiro dia de exames [91].

Neste caso de estudo foram ainda implementados outros dispositivos e sistemas que envolveram a utilização de outras tecnologias de comunicação, os quais, ainda assim, foi possível integrar na mesma plataforma IoT e garantir que o objetivo principal se mantém cumprido, apesar do escalar diversificado da rede [91].

Acrescentando às considerações mencionadas, é importante ainda referir que a componente de atuação deve também ser integrada da mesma forma nas redes IoT para que a intervenção humana se torne mínima e para que o sistema se torne o mais eficiente e eficaz possível, uma vez que esta componente tem um forte impacto nos parâmetros monitorizados pelos nós sensores. Para isto, é necessário considerar os atuadores no plano de implementação IoT e verificar se é possível a sua integração e automatização, seja de forma nativa ou através de alterações dos equipamentos.

O desenvolvimento de dispositivos em ambiente educativo permite que estes sejam criados de raiz com um objetivo e com um cenário de implementação previamente definidos, o que pode não ser a opção mais viável para uma implementação de grande escala, mas resulta em vantagens educativas interessantes para a obtenção de conhecimento prático.

4. Dispositivos e Plataformas LoRa

Analisando as diferentes componentes que integram as redes LoRa, observa-se que cada uma delas possui uma função distinta e que mesmo entre componentes de igual função existem diferenças significativas. Os dispositivos LoRa que integram a componente física da rede e que comunicam os seus dados utilizando o formato físico do sinal LoRa, são representados pelos nós sensores e pelas *gateways*. As plataformas LoRa que integram a rede protocolar e que processam os dados através de protocolos de comunicação são representadas pelos servidores de rede e pelos servidores aplicativos. Neste capítulo serão apresentados diferentes módulos e dispositivos LoRa existentes no mercado, bem como as diversas plataformas que existem para diferentes tipos de implementações IoT. A análise das características destas componentes contribui para um melhor planeamento da implementação LoRa e para uma melhor adequação à aplicação e ambiente IoT em questão.

4.1 Dispositivos LoRa

Os nós sensores e as *gateways* distinguem-se principalmente pela capacidade do seu módulo de comunicação, sendo que os nós sensores tipicamente possuem um único canal de comunicação e as *gateways* podem possuir até 64 canais, o que lhes permite comunicar com diversos nós sensores em simultâneo. Outra diferença significativa entre estes dispositivos relaciona-se com a potência de emissão, sendo que os nós sensores são tipicamente alimentados por baterias e possuem menor disponibilidade energética, o que não lhes permite recorrer a potências e ganhos de transmissão elevados. Contrariamente aos nós sensores, as *gateways* possuem potências de transmissão e alcances maiores que contribuem para uma maior abrangência da rede, apesar desta capacidade não significar diretamente um maior alcance de comunicação bidirecional. Além da comunicação LoRa, as *gateways* podem ainda possuir outras interfaces de rede, como o WiFi, *Ethernet* ou comunicação móvel (4G/5G), de forma a ligar-se à Internet e transmitir a informação da rede física para os servidores.

4.1.1 Módulos de comunicação LoRa

Os módulos de comunicação LoRa são parte integrante de todos os dispositivos LoRa, sendo que estes têm diferentes funções consoante o tipo de dispositivo no qual será integrado. Nesta secção são apresentados três módulos de comunicação com diferentes tipos de aplicação: Nós sensores comerciais, nós sensores de desenvolvimento e *gateways*.

Tipicamente, os nós sensores comerciais oferecem maior robustez e estabilidade do que os dispositivos de desenvolvimento, por terem uma qualidade de construção superior, por possuírem componentes mais adequados à comercialização e por se investir na eficácia e na eficiência destes produtos. Isto aplica-se também ao módulo de comunicação LoRa utilizado, uma vez que estes utilizam módulos como o SX1262 ou o SX1276, que só por si não permitem uma comunicação direta. Estes módulos são integrados em circuitos, juntamente com outros componentes eletrónicos, para formar um dispositivo comercializável, seja este um nó sensor ou um módulo de desenvolvimento capaz de comunicar com placas de desenvolvimento, como um Arduino ou um ESP32. O módulo SX1262 possui uma potência de transmissão de 22 dBm, uma sensibilidade de até -148 dBm, capacidade de transmissão máxima e programável de 62.5 kbps e um consumo corrente de apenas 4.2 mA no seu processo de receção de mensagens [92]. O módulo SX1276 possui uma potência de transmissão de 20 dBm, uma sensibilidade de até -148 dBm, capacidade de transmissão máxima e programável de 300 kbps e um consumo de corrente de 9.9 mA no seu processo de receção de mensagens [93]. O SX1262 é uma versão mais atualizada e otimizada do SX1276 com menor consumo de corrente, como é possível observar na Tabela 4.

Tabela 4: Comparação entre os módulos de comunicação LoRa SX1262 e SX1276 [92], [93]

Característica	SX1262	SX1276
Potência de transmissão [dBm]	22	20
<i>Link Budget</i> [dBm]	170	168
Sensibilidade [dBm]	-148	-148
Capacidade de transmissão [kbps]	62.5	300
Consumo energético [mA]	4.2	9.9

Os nós sensores de desenvolvimento possuem tipicamente módulos de comunicação LoRa com circuitos próprios, mais fáceis de integrar e prontos a utilizar, como é o caso do RFM95. Este tipo de módulos possuem circuitos de alimentação e comunicação integrados, o que lhes permite comunicar diretamente com placas de desenvolvimento através de protocolos de comunicação de *hardware*, como o SPI e reduzir os custos de desenvolvimento e pesquisa. Apresentado na Figura 23, o RFM95 é baseado no módulo SX1276 e possui as mesmas características de comunicação LoRa, sendo uma excelente opção para o desenvolvimento de protótipos e testes da tecnologia [94].



Figura 23: Módulo de comunicação LoRa RFM95 [94].

O SX1301 é um módulo de comunicação LoRa de alta capacidade e maior complexidade desenvolvido para *gateways*. A principal diferença entre este e os módulos anteriormente referidos é o número de canais de comunicação, sendo que o SX1301 possui diversos canais otimizados para a gestão de grandes volumes de dados, provenientes de diversos dispositivos em simultâneo. Este tipo de módulo de comunicação é o principal componente das *gateways* e é crucial para permitir que estas sejam o ponto de acesso dos diversos dispositivos LoRa existentes na rede. O SX1301 tem uma sensibilidade de -142 dBm, capacidade para operar com valores de SNR negativos e 10 canais de comunicação para receção de dados. Desses 10 canais de comunicação, existem 8 canais a operar com uma largura de banda de 125 kHz e que suportam a utilização de múltiplos fatores de dispersão, 1 canal capaz de operar com larguras de banda de 250 kHz ou 500 kHz, utilizando apenas um fator de dispersão e 1 canal para comunicação FSK [95].

4.1.2 Nós Sensores LoRa

O baixo consumo de energia e a bateria para alguns anos de funcionamento são características tipicamente comuns a todos os nós sensores LoRa e por isso, torna-se mais provável a eventual remoção ou troca do dispositivo em si, do que a troca de baterias. Assim sendo, a escolha do nó sensor a implementar na rede LoRa recai sobre outros fatores como o tipo de sensor, a frequência de operação, a qualidade de construção e o custo. O tipo de sensor e a qualidade de construção deverão ser estar adequados ao cenário de implementação, tendo em conta a precisão e a sua capacidade de pleno funcionamento nas condições a que estará exposto. A frequência de operação deverá estar de acordo com a sua localização e o custo de cada nó sensor deverá ser especialmente reduzido. Este custo tem um enorme impacto no plano de implementação de uma rede LoRa ou de qualquer outra rede de sensores, visto que facilmente se ultrapassam as dezenas ou centenas de nós sensores necessários para cobrir todo um ambiente de implementação, o que se pode traduzir em custos elevados da mesma.

Nesta secção, são apresentados alguns dispositivos comerciais com função de nó sensor numa rede LoRa, nomeadamente, o dispositivo ERS2 CO2, utilizado no caso de estudo da universidade de Oulu, o dispositivo Aranet4 PRO, utilizado no caso de estudo da universidade de Coruña e o dispositivo Dragino LDS02, que será implementado na rede LoRa do *campus* de Setúbal do IPS.

O nó sensor ERS2 CO2 da Elsys é um sensor de níveis de CO₂, temperatura, humidade, luminosidade e movimento, com capacidade de comunicação LoRa e alimentado por duas pilhas AA de 3.6 Volts que fornecem uma autonomia de funcionamento de até 10 anos. O dispositivo é configurável através de uma aplicação móvel e comunicação NFC ou através do envio de comandos por *downlink* na rede LoRa. Existe ainda uma componente visual do tipo semáforo, que indica o nível atual de CO₂, através da sua apresentação por cor e por intervalos de valores configuráveis. Na Figura 24 é possível observar uma representação do nó sensor ERS2 CO2 [96].



Figura 24: Nó sensor ERS2 CO2 da Elsys [96].

O nó sensor Aranet4 PRO da Aranet é um sensor de níveis de CO₂, temperatura, humidade e pressão atmosférica, com capacidade de comunicação LoRa e alimentado por duas pilhas AA que fornecem uma autonomia de funcionamento de até 10 anos. O dispositivo é configurável através de uma aplicação móvel e comunicação *Bluetooth* ou por envio de comandos a partir das *gateways* da fabricante ou ainda por *downlink* na rede LoRa. Existe também uma componente visual do tipo semáforo, que indica o nível atual de CO₂, através da sua apresentação por cor e por intervalos de valores configuráveis. Na Figura 25 é possível observar uma representação do nó sensor Aranet4 PRO [97]. Na Tabela 5 estão descritas as especificações mecânicas dos nós sensores ERS2 CO2 e Aranet4 PRO, juntamente com as suas capacidades de comunicação LoRa.



Figura 25: Nó sensor Aranet4 PRO da Aranet [97].

Tabela 5: Especificações dos nós sensores ERS2 CO2 e Aranet4 PRO [96], [97]

Especificações	ERS2 CO2	Aranet4 PRO
Dimensões [mm]	76,4 x 76,4 x 22.5	71 x 71 x 24
Classificação IP	IP20	
Temperatura (°C) e Humidade de operação (%)	0 a 50 °C 0 a 85 %	
Alimentação e Autonomia	2x Pilhas AA / Até 10 anos	
Protocolo e Classe de comunicação	LoRaWAN v1.0.4 – Classe A	
Frequências de comunicação	EU868, IN865, US915, AU915, AS923, KR923, HK923	EU868, US920, AS923, JP923, KR923
Potência de transmissão [dBm]	14	
Link budget [dB]	137dB (SF7) a 151 dB (SF12)	-----

O dispositivo LDS02 da Dragino é um sensor magnético com capacidade de comunicação LoRa, desenvolvido para a deteção de abertura e fecho de portas, sendo alimentado por duas pilhas AAA que fornecem autonomia para um envio entre 16 000 e 70 000 mensagens. O dispositivo deteta o estado de abertura/fecho da porta e envia uma mensagem com o valor desse estado, o tempo decorrido entre alterações de estado e um contador de alterações. Este possui ainda a funcionalidade nativa de alarme, enviando uma mensagem específica caso a porta se abra numa janela de tempo pré-definida, sendo que este tipo de configurações é feito através do envio de comandos por *downlink* na rede LoRa [98]. Na Figura 26 é possível observar uma representação do nó sensor LDS02 e na Tabela 6 encontram-se descritas todas as especificações do mesmo.



Figura 26: Nó sensor LDS02 da Dragino [98].

Tabela 6: Especificações do nó sensor LDS02 da Dragino [98]

Especificações	Dragino LDS02
Dimensões [mm]	40 x 15 x 15
Alimentação	2x AAA LR03
Autonomia [uplinks]	16,000 ~ 70,000
Módulo LoRa	SX1262
Protocolo e Classe de comunicação	LoRaWAN v1.0.3 – Classe A
Temperatura de operação (°C)	-10 a 50 °C

4.1.3 Gateways LoRa

A escolha da *gateway* a implementar na rede LoRa baseia-se principalmente no tipo de acesso à Internet e no tamanho da rede, em termos de área abrangida e de número de dispositivos. O acesso à Internet ou a uma rede local pode ser feito através de *Ethernet*, WiFi ou comunicações móveis, o que implica que a *gateway* tenha de possuir a interface de comunicação que melhor se adequa ao cenário de implementação. Em relação ao tamanho da rede, existem *gateways* com diferentes potências de transmissão e diferentes capacidades de gestão de dados e dispositivos, sendo que tipicamente as *gateways* com maior potência de transmissão, possuem também mais robustez para operar no exterior, maior número de canais de transmissão e geolocalização integrada.

Nesta secção são apresentados alguns dispositivos comerciais com função de *gateway* numa rede LoRa e em conformidade com as normas europeias, nomeadamente, o dispositivo Multitech Conduit IP67 Base Station, semelhante ao utilizado no caso de estudo da Universidade de Oulu, o dispositivo Aranet PRO Base Station, utilizado no caso de estudo da Universidade de Coruña e o dispositivo Kerlink Wirnet iStation, que será implementado na rede LoRa do *campus* de Setúbal do IPS.

O dispositivo Multitech Conduit IP67 Base Station é uma *gateway* LoRa desenhada para implementação exterior e resistente a condições ambientais adversas com classificação IP67. Este dispositivo configurável possui conectividade *Ethernet*, WiFi e 4G para se ligar a um servidor de rede definido pelo utilizador ou ao servidor de rede da fabricante (LENS), sendo que permite ainda receber dados de dispositivos *Bluetooth* e fornecer a sua localização através de suporte para diferentes sistemas de geolocalização. A comunicação LoRa é realizada através de 8 canais em modo *half-duplex*, com uma potência de transmissão máxima de 27 dBm para o intervalo de frequências entre 869.4 e 869.65 MHz e de 14 dBm para as restantes, sendo que esta *gateway* encontra-se tipicamente equipada com uma antena de ganho 4.5 dBi. A *gateway* é alimentada através de PoE com valores de tensão entre 37 e 57 VDC [99]. Na Figura 27 é possível observar uma representação da *gateway* Multitech Conduit IP67 Base Station.



Figura 27: *Gateway* Conduit IP67 Base Station da Multitech [99].

O dispositivo Aranet PRO Base Station é uma *gateway* LoRa desenhada para implementação interior e ligação exclusiva aos dispositivos da fabricante. Este dispositivo possui conectividade *Ethernet* e WiFi para se ligar ao servidor de rede da fabricante (Aranet Cloud), sendo que é também possível comunicar os dados através de MQTT, Modbus ou BACnet, diretamente da *gateway*. A comunicação LoRa é realizada através de 3 canais, com uma sensibilidade de -127 dBm e alcance máximo de 3 km, sendo que a *gateway* é alimentada através de PoE com 6W (12 V e 0.5 A). Na Figura 28 é possível observar uma representação da *gateway* Aranet PRO Base Station [100]. Na Tabela 7 estão descritas todas as especificações das *gateways* Multitech Conduit IP67 Base Station e Aranet PRO Base Station.



Figura 28: Gateway Aranet PRO Base Station [100].

Tabela 7: Especificações das gateways Multitech Conduit IP67 e Aranet PRO [99], [100]

Especificações	Multitech Conduit IP67	Aranet PRO
Dimensões [mm]	262 x 91 x 257	170 x 107 x 26
Classificação IP	IP67	IP42
Temperatura (°C) e Humidade de operação (%)	-45 – 70 °C -----	0 – 45 °C 0 – 90 %
Alimentação	37 – 57 VDC (> 25 W)	12 V (6 W)
Interfaces de comunicação	<i>Ethernet</i> , WiFi, 4G e <i>Bluetooth</i>	<i>Ethernet</i> , WiFi
Canais de comunicação LoRa (EU868)	8 canais <i>Half-Duplex</i>	3 canais
Alcance [km]	Até 20	3

O dispositivo Kerlink Wirnet iStation é uma *gateway* LoRa desenhada para implementação exterior e resistente a condições ambientais adversas com classificação IP67 e ISO9001. Este dispositivo configurável possui conectividade *Ethernet* e 4G para se ligar ao servidor de rede definido pelo utilizador e fornece a sua localização através do GPS integrado. A comunicação LoRa é realizada através do módulo SX1301 mencionado anteriormente e por isso, esta *gateway* possui todas as suas características de comunicação. A potência de transmissão desta *gateway* é configurável para valores entre 5 e 27 dBm, sendo que esta encontra-se tipicamente equipada com uma antena de ganho 3 dBi e é alimentada através de PoE com 30W (48 V e 0.625 A) [101]. Na Figura 29 é possível observar uma representação da *gateway* Kerlink Wirnet iStation e na Tabela 8 estão descritas todas as especificações da mesma.



Figura 29: Gateway Kerlink Wirnet iStation [101].

Tabela 8: Especificações da gateway Kerlink Wirnet iStation [101]

Especificações da gateway Kerlink Wirnet iStation	
Dimensões [mm]	265 x 165 x 100
Classificação IP	IP67
Temperatura (°C) e Humidade de operação (%)	-40 – 60 °C 0 – 95 %
Alimentação	48 VDC (30 W)
Interfaces de comunicação	<i>Ethernet e 4G</i>
Canais de comunicação LoRa (EU868)	8ch RX (125 kHz, multi SF) 1ch RX (250kHz ou 500kHz, mono SF) 1ch RX (FSK) + 1ch TX
Potência de transmissão	Configurável entre 5 e 27 dBm
Sensibilidade	- 141 dBm (SF12)

4.2 Plataformas LoRa

As plataformas IoT e LoRa permitem gerir dados e dispositivos de forma mais centralizada e tipicamente dividem-se entre servidores de rede e servidores aplicativos. Atualmente, muitos servidores de rede também já possuem uma componente aplicacional e de visualização de dados, o que significa que a escolha da plataforma ou das plataformas a utilizar irá recair sobre as diferentes vantagens que cada uma oferece e sobre as necessidades e o tipo de implementação do utilizador. As principais características diferenciadoras que definem uma plataforma são o formato de infraestrutura, podendo esta estar implementada na *cloud* ou de forma local e o licenciamento, sendo que estas podem ser *open-source* e gratuitas ou de carácter reservado e licenciado. A compatibilidade entre servidores é tipicamente garantida através de integrações bem definidas entre eles ou através de protocolos de comunicação como, por exemplo, o MQTT.

4.2.1 Servidores de rede LoRa

Os servidores de rede anteriormente mencionados como o LENS da Multitech e o Aranet Cloud da Aranet são servidores proprietários e exclusivos aos dispositivos da fabricante respetiva e não permitem a integração de dispositivos de outros fabricantes. Ainda assim, existem outros servidores de rede que oferecem compatibilidade com todo o tipo de dispositivos LoRa e que possuem diferentes formatos consoante a implementação pretendida. Nesta secção são apresentados o servidor de rede *cloud* da *The Things Network* (TTN) e o servidor de rede local ChirpStack, como alternativas *open-source* e gratuitas. É ainda apresentado o Actility Thingpark Enterprise como um servidor dependente de licenciamento, como forma de comparação de funcionalidades com os servidores gratuitos.

O The Things Network é uma plataforma IoT composta por diversos servidores de rede ajustados às necessidades dos utilizadores, mas com o objetivo principal de oferecer um servidor *open-source* e de criar uma comunidade global de redes LoRaWAN e de partilha de recursos com base nesta plataforma. Os servidores de rede que compõem o TTN baseiam-se numa arquitetura denominada de The Things Stack, que oferece diversos serviços e ferramentas de gestão de rede LoRaWAN e de dispositivos LoRa. O The Things Stack Cloud é um servidor de rede *cloud* focado para empresas e implementações de nível comercial, com base em licenças e independência de infraestrutura física, sendo que toda a gestão da rede é processada nos servidores da TTN. Em alternativa, o The Things Stack Enterprise permite que o utilizador implemente o servidor de rede na sua própria infraestrutura [102].

O The Things Stack Open Source é a versão mais básica e totalmente aberta à comunidade, permitindo aos utilizadores adaptar toda a estrutura do servidor às suas necessidades. O The Things Stack Sandbox (TTSS) é o servidor de rede mais utilizado pela comunidade LoRaWAN, oferecendo uma versão gratuita e limitada da arquitetura The Things Stack para implementações não comerciais, educativas e experimentais. Este servidor de rede faz uso da capacidade de comunicação com múltiplos dispositivos das *gateways* e possibilita a partilha deste recurso com a comunidade, permitindo que qualquer utilizador utilize o seu dispositivo LoRa através das redes LoRaWAN baseadas neste servidor. A sua interface é representada por um painel de controlo de monitorização da rede, onde é possível observar todas as informações e estados da rede, nós sensores e *gateways*. Este servidor oferece ainda a integração direta dos dados com diversos servidores aplicativos, como o AWS IoT ou o Azure IoT Hub, a integração através de protocolos de comunicação, como o MQTT ou HTTP e ainda a configuração de *webhooks* como forma de notificação de eventos e de envio de dados para outros serviços aplicativos, como o TagoIO ou o TTN Mapper [102].

A integração das *gateways* no servidor permite analisar o estado das mesmas, observar os dados dos dispositivos que com ela comunicam, criar acessos à *gateway*, tanto através da plataforma, como de APIs e ajustar as diversas configurações relacionadas com o protocolo LoRaWAN. A integração dos nós sensores no servidor é realizada através de aplicações da plataforma e cada uma pode conter os seus próprios dispositivos, formatações de *payloads*, integrações, credenciais de acesso e configurações. Nestas aplicações é possível observar a informação enviada pelos nós sensores associados e alterar diversas configurações de comunicação dos mesmos. Os *payloads* recebidos e os *payloads* que o servidor envia para os nós sensores podem ser formatados diretamente no servidor de rede, através de serviços como o CayenneLPP ou GRPC ou através de *scripts* Javascript, de forma a apresentar os dados da forma mais adequada. Dentro de cada aplicação são também criadas as integrações e as credenciais necessárias para cada integração ou acesso. Na página de cada nó sensor é possível observar os dados enviados por este, as informações de cada comunicação de dados, incluindo a intensidade do sinal e o SNR, simular o envio e receção de dados, definir a sua localização, formatar o *payload* e configurar a comunicação LoRaWAN [102]. Na Figura 30 é possível observar a *dashboard* e visão geral de uma aplicação criada no TTSS.

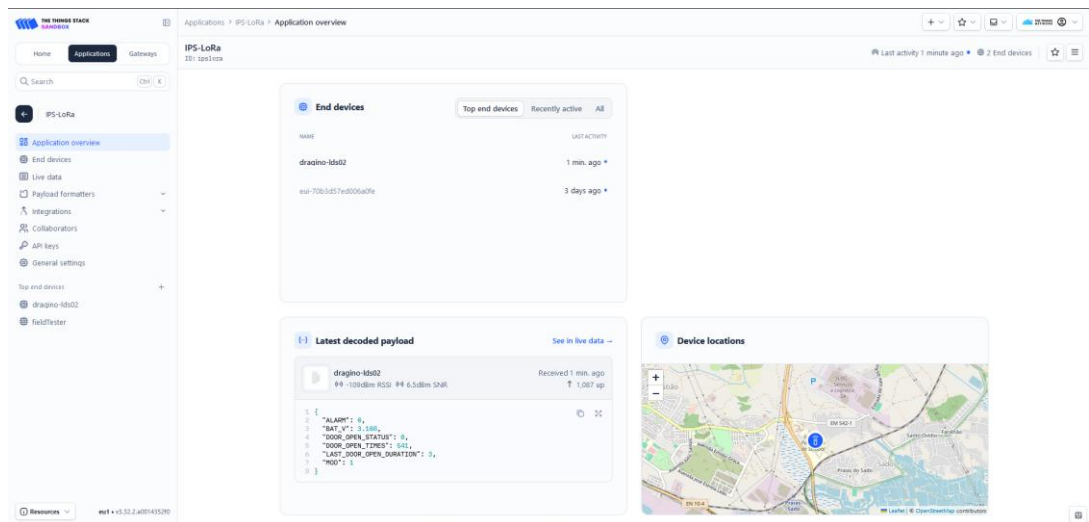


Figura 30: Visão geral da aplicação IPS-LoRa - TTSS.

Em alternativa ao The Things Stack, o servidor de rede ChirpStack é igualmente *open-source* e de utilização gratuita, mas necessita de ser implementado em infraestrutura própria e em ambientes como Docker, Ubuntu ou Raspberry Pi. A plataforma permite que o utilizador tenha controlo total sobre todas as componentes da rede e através da sua interface *web* permite gerir a rede e os seus dispositivos e realizar integrações com as principais ferramentas e serviços existentes. As suas restantes funcionalidades e configurações são semelhantes às do TTSS, embora o ChirpStack se foque mais na criação de redes privadas e na configuração mais técnica das mesmas, o que acrescenta alguma complexidade à implementação, em contraste com a solução mais simples e intuitiva da TTSS [104]. Na Figura 31 é possível observar a *dashboard* do ChirpStack e constatar a menor intuição do servidor de rede em comparação com o TTSS.

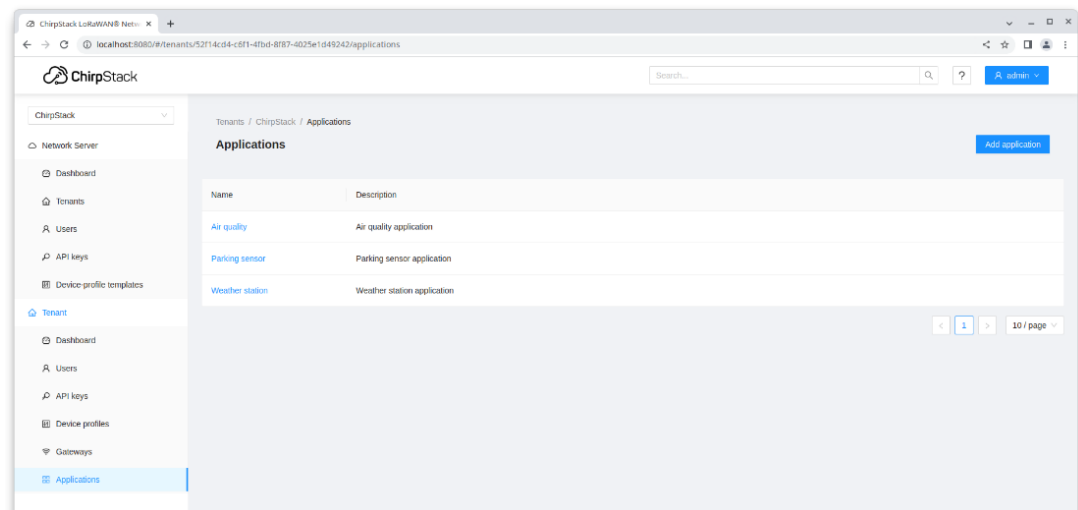


Figura 31: *Dashboard* – Chirpstack [104].

O servidor de rede Thingpark Enterprise da Actility é uma alternativa dedicada ao setor comercial e industrial de larga escala, com maior suporte ao cliente e construção personalizada à implementação do mesmo. À semelhança do ChirpStack, este servidor é totalmente configurável e adaptável às necessidades da implementação, mas, no entanto, o grau de personalização e a flexibilidade são maiores. Esta solução comercial depende da aquisição de licenças, mas oferece uma componente de serviço de cliente mais alargada, através de atualizações, manutenção e suporte a todo o servidor de rede implementado e componente aplicacional personalizada [105]. Na Figura 32 é possível observar o painel de controlo do Thingpark Enterprise.

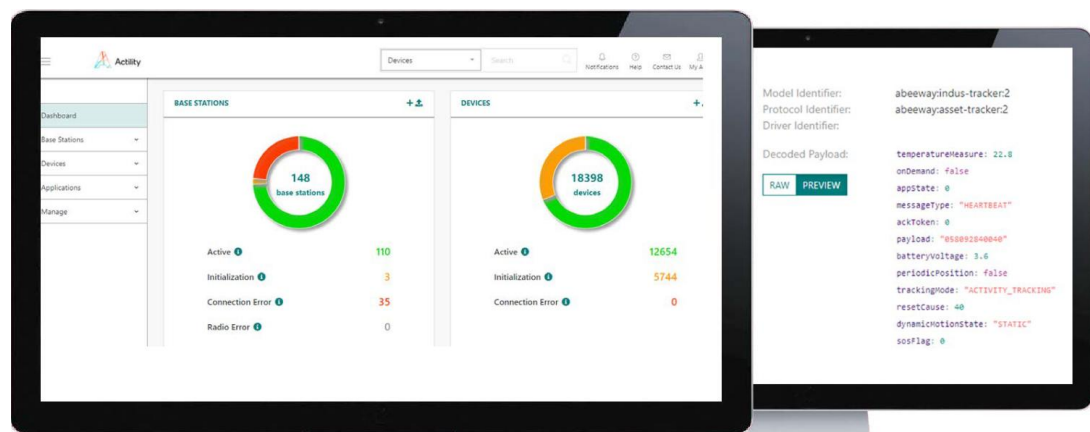


Figura 32: Painel de controlo – Thingpark Enterprise [105].

4.2.2 Servidores aplicacionais

Os servidores aplicacionais são responsáveis por apresentar os dados aos utilizadores e por permitirem a interação com a rede através de interfaces desenhadas para o utilizador final. Visto que estes servidores possuem uma forte componente de visualização de dados, as empresas que os desenvolvem investem continuamente numa maior acessibilidade, personalização e interatividade da informação que é apresentada aos utilizadores. Os servidores mencionados nos casos de estudo foram especialmente desenvolvidos para uma área de atividade, como é o caso do ThingWorx para a área da indústria e da Home Assistant para a automação residencial, ainda que existam outros servidores aplicacionais mais abrangentes e potencialmente mais personalizáveis, como é o caso do TagoIO.

O servidor aplicacional ThingWorx permite desenvolver soluções IoT de grande escala, com especial foco na integração de dispositivos e sistemas industriais como ERPs, CRMs e SCADA. À semelhança do Thingpark Enterprise, este servidor implica a compra de licenças, mas oferece um importante suporte ao tipo de aplicações críticas que o setor da indústria possui [106]. Na Figura 33 é possível observar um exemplo do tipo de *dashboard* que é possível criar com o ThingWorx.

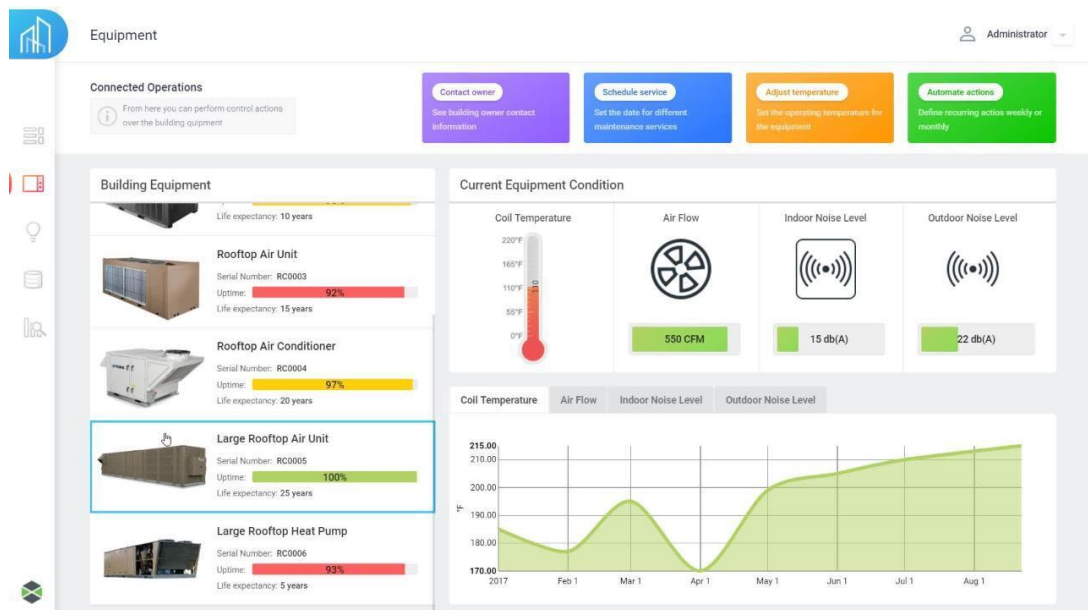


Figura 33: Exemplo de *dashboard* – ThingWorx [106].

O Home Assistant é um servidor aplicativo *open-source* e gratuito focado no desenvolvimento fácil e intuitivo de sistemas IoT de automação residencial. O seu formato local garante uma maior privacidade e limita o acesso à informação, sendo que este servidor consegue ainda operar em dispositivos de baixo poder de processamento como um Raspberry Pi. A sua *dashboard* intuitiva e personalizável permite que o utilizador construa a sua aplicação da forma que mais se adequa às suas necessidades e preferências [107]. Na Figura 34 é possível observar um exemplo de uma *dashboard* criada no Home Assistant e composta por diversos sensores distribuídos pela residência.

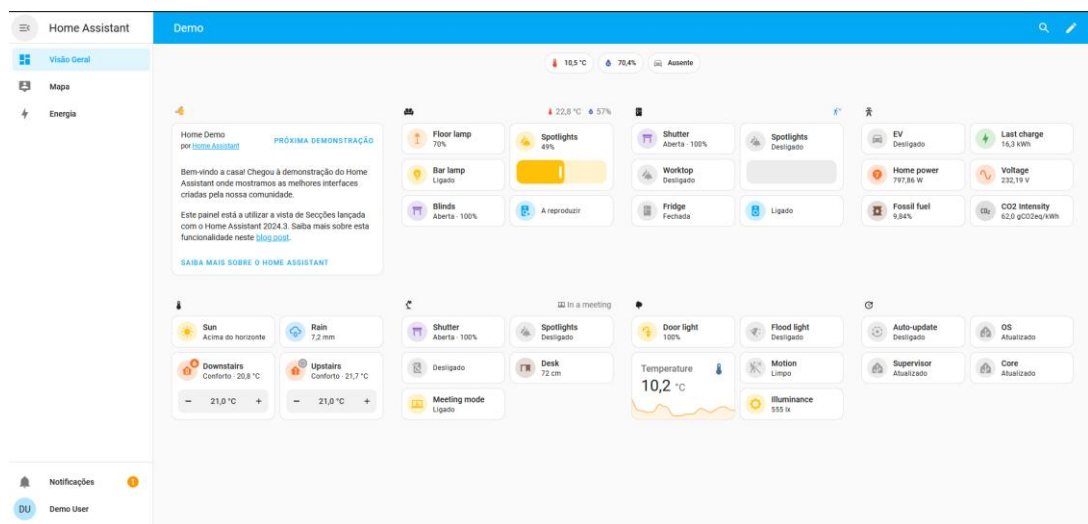


Figura 34: Exemplo de *dashboard* – Home Assistant [107].

O TagoIO é também um servidor aplicacional *open-source* e gratuito, mas oferece mais funcionalidades e abrange um maior número de aplicações e dispositivos IoT. A sua *dashboard* é totalmente personalizável e é possível conjugar diversos objetos de visualização de dados, como gráficos ou tabelas com *scripts* avançados de automação, capazes de interagir em tempo real com os dados [108]. Na Figura 35 é possível observar um dos muitos *templates* de *dashboard* que o TagoIO oferece, sendo que esta é composta por botões programáveis e diferentes tipos de visualização e histórico de dados.



Figura 35: Exemplo de *dashboard* – TagoIO [108].

A componente de desenvolvimento do TagoIO apresenta ainda diversas funcionalidades que permitem programar tarefas ou gerir e armazenar dados e ficheiros do sistema IoT. Na Figura 36 é possível observar as principais opções agrupadas no menu do servidor, nomeadamente, a página inicial, os dispositivos, as entidades, os ficheiros, as análises, as ações, o acesso, os utilizadores e a execução. A página inicial apresenta um resumo dos recursos integrados, configurados e utilizados até ao momento. A opção dos dispositivos permite adicionar conectores, os quais representam integrações diretas com servidores de rede, ligações específicas a dispositivos ou ligações através de protocolos de comunicação. As entidades constituem a componente própria de armazenamento do TagoIO, nas quais é possível armazenar e gerir os dados de forma semelhante a outras tecnologias de base de dados. A opção de ficheiros permite armazenar todo o tipo de ficheiros relacionados com os dispositivos, com a gestão de dados ou com as *dashboards* criadas. A opção de análises permite criar *scripts* de análise e processamento de dados e implementá-los no sistema IoT. As ações representam as automações criadas através da execução programada dos *scripts* mencionados. A opção de acesso permite que outros utilizadores façam a gestão das componentes criadas através de acessos controlados. Os utilizadores podem ser criados para fornecer um tipo de acesso específico à *dashboard*, limitando ou autorizando o acesso aos dados e à interação. A execução permite partilhar a *dashboard* criada em formato de aplicação *web* para diferentes tipos de dispositivos [108].

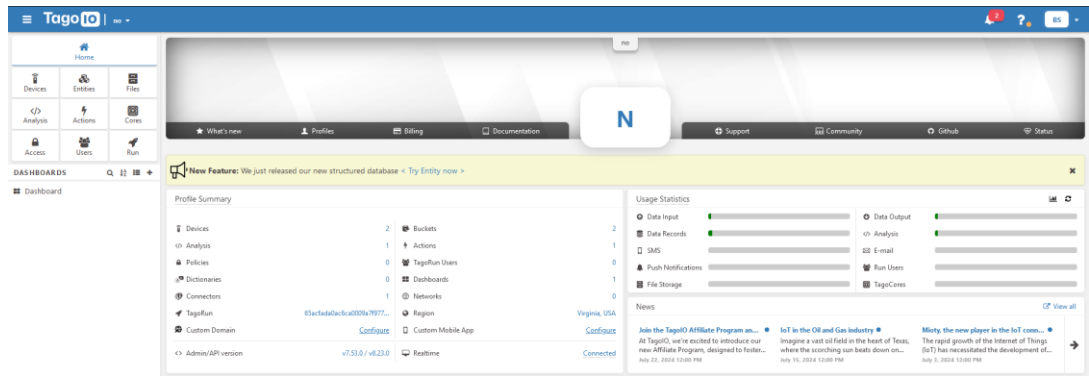


Figura 36: Painel de controlo – TagoIO [108].

5. Proposta de implementação IoT – *Campus* de Setúbal – IPS

A formulação de uma proposta adequada e o planeamento de uma implementação IoT são processos fundamentais para o sucesso e para a sustentabilidade da implementação, pois permitem adaptá-la às necessidades dos utilizadores e às condições do local de implementação, tornando-a mais vantajosa, eficiente e simples de implementar. O planeamento de uma implementação IoT inicia-se pela caracterização do local de implementação e pela identificação dos objetivos e resultados esperados da mesma. A caracterização do local de implementação e a identificação dos objetivos permitem aferir previamente algumas das condições que poderão ter impacto no tipo de rede que irá ser implementada, como o tipo de geografia do local, o alcance da comunicação ou a densidade e tipo de dispositivos. Inseridas no seu planeamento, esta proposta de implementação apresenta projetos em atual desenvolvimento e possíveis aplicações desenhadas para *campus* de Setúbal do IPS. A arquitetura proposta foca-se em garantir a escalabilidade e a acessibilidade da rede, implementado uma base sólida da mesma com capacidade largamente superior ao que será implementado inicialmente no decorrer desta dissertação. Neste capítulo será apresentado o planeamento da implementação IoT, juntamente com a caracterização do *campus* de Setúbal do IPS e com uma análise da geografia ao seu redor, sendo ainda apresentada a arquitetura da rede IoT proposta, os projetos em desenvolvimento e possíveis aplicações desenhadas para o *campus*.

5.1 Planeamento da implementação IoT

A rede IoT a implementar terá como principal local de implementação o *campus* de Setúbal do IPS, sendo que esta rede irá abranger ainda outras áreas circundantes e possibilitar o envolvimento de outras instituições e infraestruturas próximas ao *campus*. O *campus* de Setúbal localiza-se num terreno inclinado com uma média de 10 metros de altitude e possui uma área aproximada de 0,258 km², constituída por diversos espaços verdes, estacionamento e 4 edifícios principais, onde decorrem a grande maioria das atividades letivas e não-letivas da comunidade académica. Visto que este se encontra próximo do centro do *campus* e é onde se localiza as infraestruturas do ramo de Eletrónica e Telecomunicações do Departamento de Engenharia Eletrotécnica (DEE-ET), assumiu-se o bloco F da Escola Superior de Tecnologia de Setúbal (ESTS) como ponto de referência, verificando-se que, nestas condições, a rede IoT terá de possuir um alcance de comunicação mínimo de 450 metros de forma a garantir uma cobertura total do *campus*, sendo que este alcance é facilmente superado pelas diversas tecnologias de comunicação e arquiteturas típicas de redes IoT.

Na Figura 37 é possível observar um mapa topográfico em duas dimensões do *campus* e verificar as variações de altitude que existem no mesmo, sendo que é possível ainda concluir que os edifícios se encontram aproximadamente à mesma altitude. Na Figura 38 é possível observar um mapa em três dimensões do *campus* e verificar que as alturas dos edifícios são também semelhantes entre si.

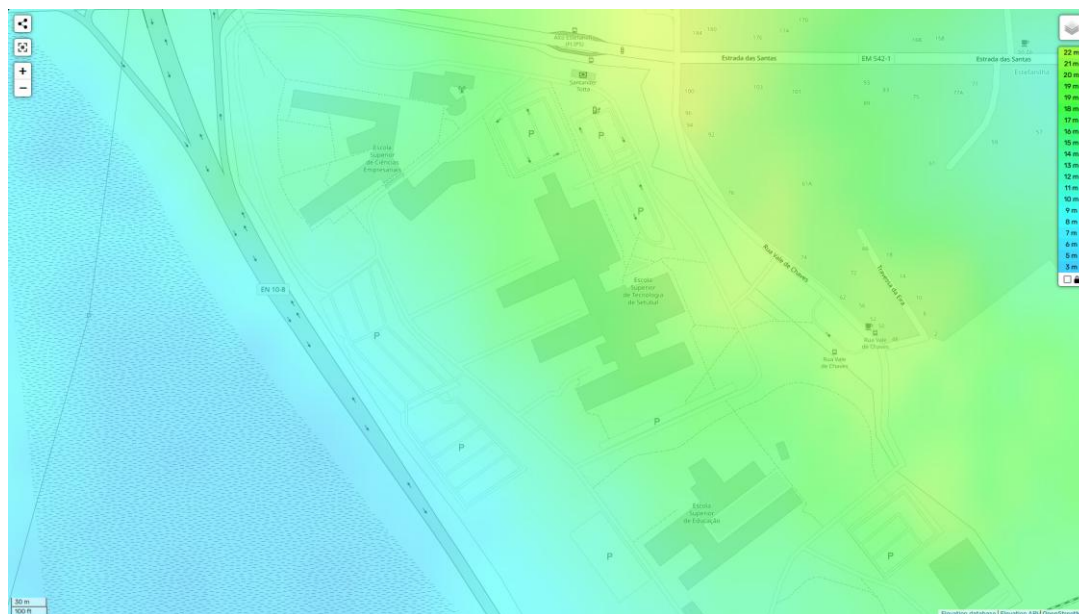


Figura 37: Mapa topográfico do *campus* de Setúbal do IPS [109].



Figura 38: Representação 3D do *campus* de Setúbal do IPS [110].

Ao redor do *campus*, existem diversas áreas e infraestruturas de interesse, como a Escola Profissional de Setúbal, a Escola Secundária Dom Manuel Martins, a Doca do Comércio de Setúbal, entre outros locais que poderão usufruir da rede IoT e contribuir para o seu enriquecimento. A cidade de Setúbal e as zonas que a envolvem, como a Arrábida, Palmela ou Azeitão, possuem diversas variações de altitude e uma geografia bastante diversificada, sendo que a Sul e a Este da cidade, a altitude é mais baixa e a área é predominantemente hídrica, enquanto que a Norte e a Oeste da cidade, é possível encontrar áreas montanhosas e de altitudes mais elevadas.

Na Figura 39 é possível observar um mapa topográfico e representativo das variações de altitude da área referida, com indicação da altitude do *campus* de Setúbal para comparação com a restante área. O clima da zona de Setúbal é relativamente moderado e tipicamente possui um impacto reduzido nas variações de atenuação de propagação.

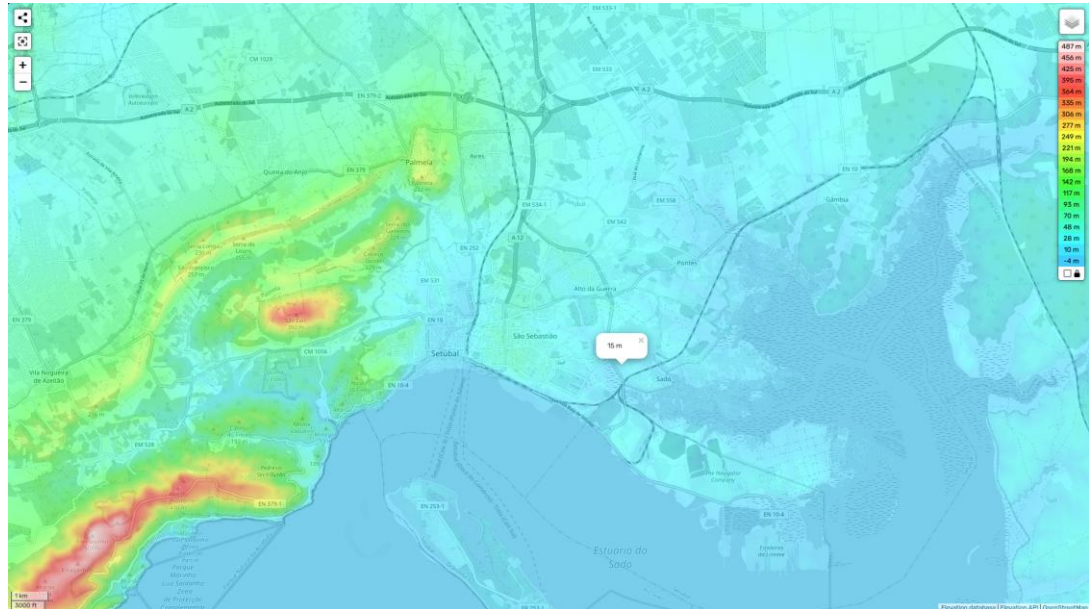


Figura 39: Mapa topográfico da região de Setúbal [109].

Em meio urbano, o alcance típico de comunicação do LoRa é de 5 km, o que permite abranger uma boa parte da cidade de Setúbal, mas não cobre a sua totalidade nem as áreas envolventes referidas anteriormente. Ainda assim, devido à geografia mista desta zona deverá ser possível realizar comunicações a distâncias superiores. Assumindo uma cobertura da rede com um raio de alcance de 10 km, já seria possível abranger as áreas referidas. Na Figura 40 e Figura 41 é possível observar possíveis coberturas da rede para raios de 5 e 10 km.

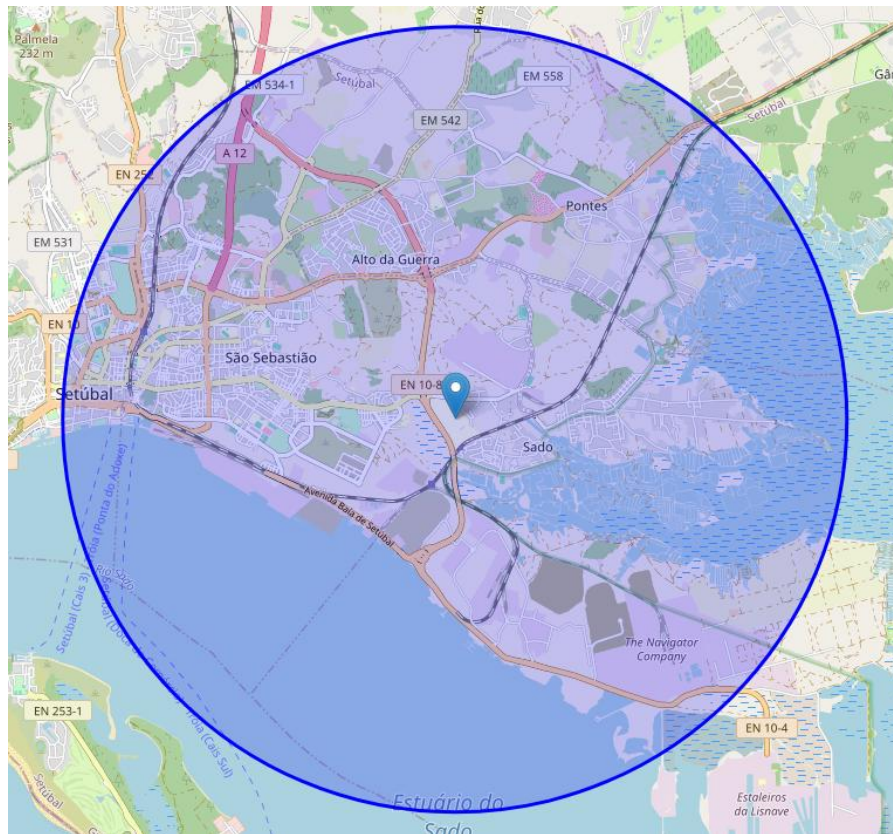


Figura 40: Exemplo de cobertura para um raio de 5 km do *campus*.

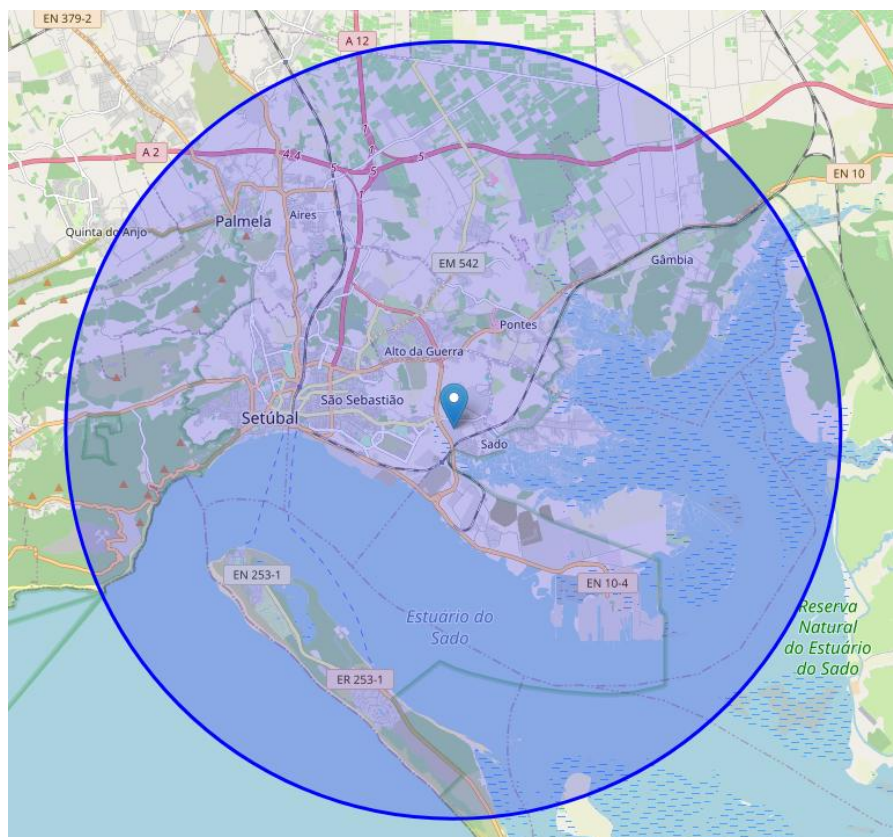


Figura 41: Exemplo de cobertura para um raio de 10 km do *campus*.

A identificação de objetivos ou necessidades para uma implementação IoT envolve a avaliação de diferentes critérios específicos da implementação e do local onde a rede IoT é implementada. Neste caso específico, de forma a cobrir o *campus* com uma rede IoT e a possibilitar a integração de outros serviços ao redor do mesmo, é necessário que a tecnologia de comunicação a utilizar seja robusta e que possua um alcance de comunicação alargado. A robustez da comunicação permite que os efeitos provocados por obstáculos ou fatores ambientais sejam reduzidos e que as falhas de comunicação sejam escassas, ao longo de todo um alcance de comunicação que permita cobrir o *campus* e futuras aplicações ao seu redor. Dada a dimensão do *campus* e o grande número de aplicações que é possível implementar, é ainda necessário garantir que, tanto a *gateway* como os servidores a utilizar, têm capacidade para gerir um grande número de dispositivos em simultâneo e que tanto o seu custo de implementação como o seu consumo energético são reduzidos. A capacidade de transmissão e a latência são características pouco relevantes neste tipo de aplicação, visto que o volume de dados a transmitir por cada nó sensor será reduzido e que o intervalo entre transmissões ultrapassa largamente a latência típica das tecnologias de comunicação de baixo custo e longo alcance, utilizadas neste tipo de redes, e em particular a tecnologia de comunicação LoRa.

Por motivos de segurança, a infraestrutura de rede do *campus* de Setúbal não permite o acesso a servidores externos nem o acesso externo à mesma, o que dificulta o acesso à rede IoT e impossibilita o acesso remoto a esta. Assim sendo, será necessário garantir que a segurança da rede do *campus* não é comprometida, mas que o acesso remoto à rede IoT é possível, para que o número de aplicações da rede não seja limitado e para que se beneficie de todas as funcionalidades que a *gateway* e o servidor de rede oferecem. É ainda necessário que o servidor de rede a utilizar ofereça diversas formas de integração com outras plataformas, seja por integração direta ou através da utilização de protocolos de comunicação específicos. Desta forma será possível utilizar diferentes servidores aplicativos em simultâneo e associar conjuntos de dispositivos a plataformas e ferramentas mais adequadas ao tipo de aplicação IoT pretendida. Apesar do enorme contributo das aplicações IoT para uma maior eficiência e eficácia dos serviços académicos e das infraestruturas do *campus*, esta implementação tem como principal objetivo, a inovação, a pesquisa e o desenvolvimento de aplicações IoT em contexto educativo. A implementação de uma rede IoT no *campus* irá permitir o desenvolvimento de novos projetos, a utilização de novas ferramentas e a aplicação prática de conceitos lecionados em diversas unidades curriculares e cursos oferecidos pelo IPS.

Uma importante consideração neste planeamento é também o desenvolvimento de documentação sólida e a garantia de manutenção da rede, para que os projetos desenvolvidos se mantenham em funcionamento e para que os futuros projetos possuam uma base de conhecimento da rede, que os permita inovar e construir novo conhecimento sobre estes.

5.2 Arquitetura da implementação IoT

Tendo em conta a análise realizada aos conceitos e tecnologias de comunicação IoT, às características do local de implementação e às necessidades e objetivos do *campus*, determinou-se que a implementação e que a sua arquitetura seja baseada numa rede IoT com comunicação LoRa e utilização do servidor de rede The Things Stack Sandbox. Será ainda utilizado o servidor aplicacional TagoIO, um protótipo para testes de comunicação (fieldTester) e um dispositivo comercial, de forma a testar diferentes utilizações da rede.

A escolha da tecnologia de comunicação LoRa deve-se à sua robustez e alcance de comunicação alargado que permitem cobrir todo o *campus*, sendo que tanto a própria tecnologia como o seu protocolo de comunicação LoRaWAN foram desenvolvidos com ênfase na comunicação simultânea com diversos dispositivos de baixo custo e baixo consumo energético. De forma a garantir uma melhor cobertura e manutenção da rede, a *gateway* LoRa terá de ser instalada num local alto, exterior, facilmente acessível pela equipa responsável pela sua manutenção e de preferência junto ao local de referência mencionado anteriormente. Dessa forma, a *gateway* estaria aproximadamente localizada no centro do *campus* e o seu acesso ficaria mais facilitado por parte dos responsáveis pela rede. A *gateway* Kerlink Wirnet iStation foi selecionada para esta implementação, devido à sua capacidade de funcionamento no exterior, à sua potência de transmissão e à sua capacidade de comunicação com diversos dispositivos em simultâneo, sendo que a *gateway* poderá ser facilmente alimentada e ligada à Internet, através de PoE no local de implementação referido. A solução encontrada para as dificuldades associadas à política de segurança da rede envolve a integração da *gateway* numa rede virtual associada à rede do *campus* e a utilização de um servidor de rede na *cloud*, como o TTSS, que permita o acesso aos dados sem a necessidade de acesso direto e local à rede. Este servidor é proposto para esta implementação por ser *open-source* e gratuito, mas também por se focar na partilha de recursos com a comunidade e por permitir a utilização da *gateway* de forma aberta sem que seja necessária a intervenção do gestor da rede. Esta última funcionalidade permite que qualquer membro da comunidade académica ou outro utilizador utilize o seu dispositivo LoRa com esta *gateway* e com este servidor de rede, sem que seja necessária a intervenção dos responsáveis da rede e sem que esse dispositivo seja agregado a sistemas próprios do instituto. Utilizando o TTSS, o servidor de rede encontra-se fora da rede do *campus* e torna-se possível realizar ligações a outros servidores, como o TagoIO, sem qualquer condicionante. Este servidor aplicacional foi selecionado devido às suas extensas funcionalidades e personalização, mas também por ser gratuito e possuir integração direta com qualquer servidor TTN.

O protótipo fieldTester foi desenvolvido de raiz com o intuito de criar um dispositivo móvel, versátil e adaptável a diferentes testes de comunicação, sendo este constituído por componentes programáveis de desenvolvimento, nomeadamente, uma placa Arduino e o módulo de comunicação RFM95 apresentado anteriormente. Este protótipo é ainda constituído por um sensor de temperatura e por um módulo *Bluetooth* que permite enviar os valores de RSSI para uma aplicação móvel que associa esses valores a coordenadas de localização. O sensor magnético de porta Dragino LDS02 pretende demonstrar a capacidade de comunicação de dispositivos fabricados com intuito comercial e o tipo de resultados possíveis de obter com este tipo de aplicação, sendo que este sensor será instalado na entrada de um local de acesso limitado e que permitirá monitorizar a abertura e fecho dessa porta.

Desta forma, define-se uma arquitetura de implementação que pretende implementar uma *gateway* de alto nível, capaz de gerir uma rede de elevada dimensão e associá-la a uma plataforma comunitária que facilite a sua utilização por parte dos utilizadores, preparando a rede IoT implementada para futuras aplicações. As restantes componentes são mais específicas à implementação em causa e ao propósito desta dissertação, sendo totalmente alteráveis e diferentes para cada projeto ou implementação IoT. Na Figura 42 é possível observar a arquitetura proposta para esta implementação e a tabela 9 apresenta os custos associados à mesma.

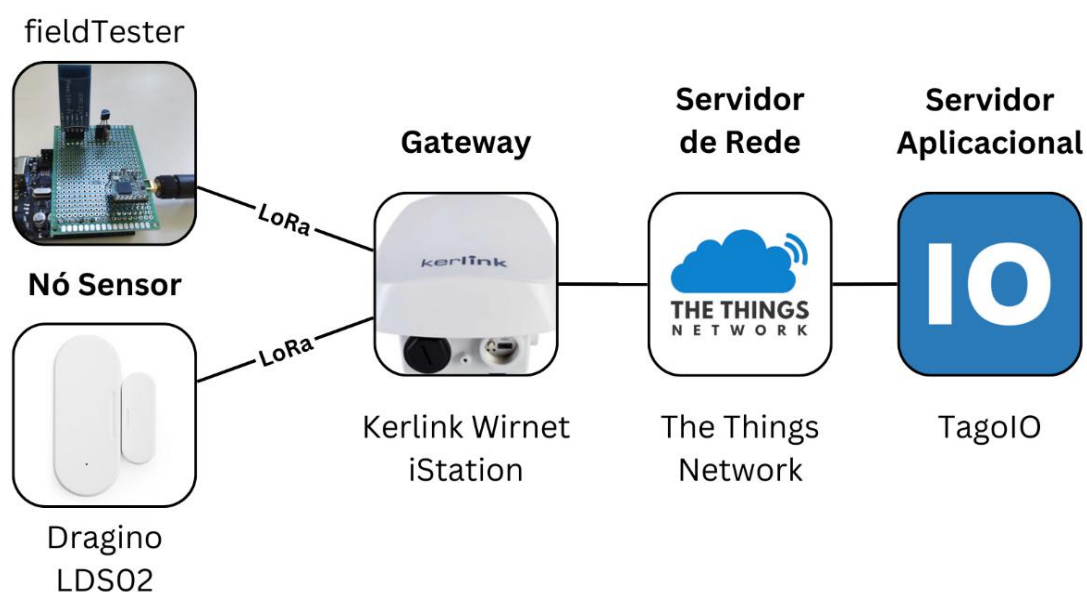


Figura 42: Arquitetura da implementação IoT no *campus* de Setúbal do IPS.

Tabela 9: Custos da implementação IoT no *campus* de Setúbal do IPS

Item		Preço (c/ IVA) [€]
Gateway	Gateway Kerlink Wirnet iStation	1 176,19€
	Antena Omnidir 868MHz 3dBi KLK03198	39,98€
	PoE TP-Link TL-POE160S 30W	20,00€
Nó Sensor comercial	Dragino LDS02	24,55€
Nó Sensor protótipo	Arduino + Placa perfurada + Ligações	25,00€
	Sensor de temperatura digital DS18B20	4,90€
	Módulo LoRa RFM95 + Antena 3dBi	11,40€
	Módulo <i>Bluetooth</i> HC-06	7,40€
	Pilha 9V + Suporte	10,00€
Custo Total		1319,42€

5.3 Aplicações IoT para o *campus*

Atualmente e entre os diversos projetos que planeiam utilizar esta rede IoT, existem dois projetos em desenvolvimento e em contexto de projeto final de curso de licenciatura e dissertação de mestrado. O projeto final denominado de ReSens envolve a implementação de uma rede de sensores de monitorização ambiental no bosque Miyawaki do *campus*, com o intuito de fornecer informações para uma atuação mais eficiente do sistema de rega do bosque. Utilizando a rede LoRa implementada no desenvolvimento desta dissertação, os nós sensores do projeto ReSens realizam a comunicação com o TTSS e este intermedia os dados através de MQTT para o servidor aplicacional local (Node-RED) instalado num Raspberry Pi. A partir do servidor aplicacional, os dados são apresentados numa aplicação *web* associada a uma base de dados baseada numa arquitetura PHP [111]. O projeto da dissertação denominado de LoRaGrid envolve a instalação de um sistema de monitorização de consumos energéticos de igual formato de comunicação, mas com recurso a um servidor aplicacional em formato *web* desenvolvido pelo aluno e especificamente para o tipo de aplicação pretendida. A comunicação MQTT é feita para um servidor *web* que hospeda uma aplicação *web* e que serve de interface de visualização de dados [112].

Ambos os projetos irão acrescentar valor à rede IoT implementada e fornecer dados sobre diferentes áreas do *campus*, contribuindo para a prova de conceito que irá ser desenvolvida e analisada nesta dissertação. Um exemplo de aplicação que contribuiria bastante para a economia de recursos e para o bem-estar da comunidade acadêmica seria a implementação de um sistema de monitorização de presença e luminosidade capaz de interagir com a iluminação dos diversos espaços do *campus*. Desta forma, seria possível ligar apenas a iluminação necessária e ajustar a sua intensidade de acordo com a presença de pessoas e a luminosidade natural. Realizando uma análise superficial do impacto desta aplicação e agregando no preço de cada sensor, os custos das restantes componentes potencialmente necessárias, como um Raspberry Pi para servidor aplicacional, conclui-se que uma implementação de 100 sensores, em que cada um tem o custo de 60€, teria um custo aproximado de 6000€. No entanto, assumindo que cada sensor estaria ligado a 5 luminárias com uma potência de 18W e um uso diário de 12h por cada, o consumo diário de 500 luminárias seria de 108kWh, o que se traduz numa despesa diária de 17,28€, quando assumido o custo de 0,16€ por kWh. Se esta implementação permitisse uma poupança de 30% no consumo energético destas luminárias, ou seja, se fosse possível obter uma poupança diária de 5,184€, seriam apenas necessários aproximadamente 3 anos e meio para que o investimento compensasse a poupança de energia. Esta janela de tempo poderá ser ainda menor quando se tem em consideração o menor desgaste e o menor consumo energético das luminárias derivado do controlo de intensidade das mesmas.

Contribuindo para a manutenção e para a inovação dos projetos recentemente desenvolvidos no contexto de *smart campus*, seria ainda interessante a inclusão futura de mais sensores e funcionalidades nestes projetos por parte de outros membros da comunidade académica. Um exemplo de novos desenvolvimentos com base em projetos já implementados seria a inclusão de sensores de presença de animais ou de qualidade do ar no bosque Miyawaki do *campus*, a qual permitiria uma análise mais completa do ecossistema e das suas variações ao longo do dia e ao longo das mudanças de estação do ano. Aproveitando o carácter de rede de sensores do projeto LoRaGrid, seria também benéfico a inclusão de sensores que monitorizem as diferentes redes de recursos existentes no *campus*, como a rede de água ou de esgotos. Para além da monitorização e criação de histórico de dados, uma das funcionalidades mais importantes a implementar seria a notificação de eventos em ambos os projetos, sendo que esta permitiria alertar em tempo real para eventuais emergências ou desperdícios de recursos nestas redes.

6. Implementação da rede IoT

A implementação da rede IoT baseia-se na proposta apresentada no capítulo 5 e no desenvolvimento, instalação e configuração das componentes apresentadas, com base em comunicação LoRa. Esta implementação permitiu realizar uma análise desta tecnologia de comunicação através da programação de um módulo de comunicação LoRa, da verificação de diferentes formas de ativação de dispositivos LoRa, da determinação da intensidade de sinal recebido no dispositivo de testes, entre outras tarefas que contribuíram para a aplicação prática da tecnologia e de conceitos de comunicação sem fios. A comunicação através do protocolo LoRaWAN foi verificada através da instalação da *gateway* e da configuração do servidor de rede, no qual os dados são decodificados e transmitidos ao servidor aplicacional responsável pelo posterior tratamento dos mesmos e pela visualização da informação. A forte componente *open-source* do dispositivo desenvolvido e das plataformas utilizadas contribuíram também para a profundidade do estudo realizado e para a atenuação dos custos e da complexidade de implementação. Neste capítulo é apresentado o desenvolvimento do dispositivo de testes, a instalação do dispositivo LoRa comercial e da *gateway*, bem como a configuração da mesma e do servidor de rede LoRa, juntamente com a integração dos dispositivos neste servidor e no servidor aplicacional, culminando na verificação do correto fluxo de dados em toda a rede.

6.1 Nós Sensores LoRa

Os nós sensores LoRa a implementar representam dois cenários de implementação completamente distintos com o objetivo de demonstrar detalhadamente o processo de comunicação LoRa programável e a simplicidade de implementação de dispositivos comerciais. O dispositivo para testes de cobertura utiliza o método de ativação ABP e foi programado para transmitir valores de temperatura como forma de demonstração simples da comunicação LoRa com o servidor de rede. O dispositivo comercial é constituído por um sensor magnético que tem como principal objetivo a verificação da abertura e fecho de portas, sendo que este utiliza o método de ativação OTAA por ser mais seguro e robusto para aplicações comerciais. Como apresentado na secção 2.3, estes métodos possuem diferentes características de ativação e estas implementações permitem verificar essas mesmas diferenças no processo de ativação e configuração dos dispositivos em contexto prático.

6.1.1 Dispositivo para testes de cobertura – fieldTester (RFM95)

O dispositivo para testes de cobertura denomina-se de “fieldTester” por ser utilizado para testes de comunicação em contexto real e em qualquer local coberto pela rede. Este dispositivo tem como função principal a comunicação de pequenos pacotes de dados com valores do sensor de temperatura, de forma a testar a capacidade, a qualidade e o alcance da comunicação em diversos pontos do local de implementação. A capacidade e qualidade de comunicação avalia-se pelo sucesso do envio de dados para a *gateway* e respetiva visualização dos mesmos no servidor de rede, juntamente com valores de qualidade de sinal, de que constitui exemplo, o SNR. O alcance da comunicação é avaliado pelo sucesso da receção de dados no dispositivo e pela determinação da intensidade do sinal recebido pelo mesmo. Através de comunicação *Bluetooth*, é possível transmitir esse valor de RSSI a uma aplicação móvel que o associa às coordenadas do dispositivo móvel utilizado, criando um ficheiro de dados exportável. Os dados do ficheiro permitem construir um mapa de cobertura associando pontos do mapa a valores de RSSI obtidos. O fieldTester utiliza uma placa de desenvolvimento Arduino e um shield adaptado a esta, constituído por um módulo de comunicação LoRa RFM95 e respetiva antena de 3 dBi, por um módulo *Bluetooth* HC-06 [113], por um sensor de temperatura digital DS18B20 [114] e por um LED indicador de *downlink*. O módulo de comunicação LoRa é alimentado através de 3.3V e comunica com o Arduino através de SPI, utilizando ainda mais 4 portas digitais para realizar as restantes comunicações necessárias. O módulo *Bluetooth* é alimentado através de 5V e realiza comunicação série com o Arduino. O sensor de temperatura digital é também alimentado por 5V e comunica com o Arduino através de uma porta digital. O LED indicador é controlado através de uma porta digital. A sua componente móvel é garantida através da alimentação do Arduino com uma pilha de 9V que oferece algumas horas de autonomia para uma utilização intervalada. Na Figura 43 é possível observar o esquemático e as ligações entre as componentes do fieldTester.

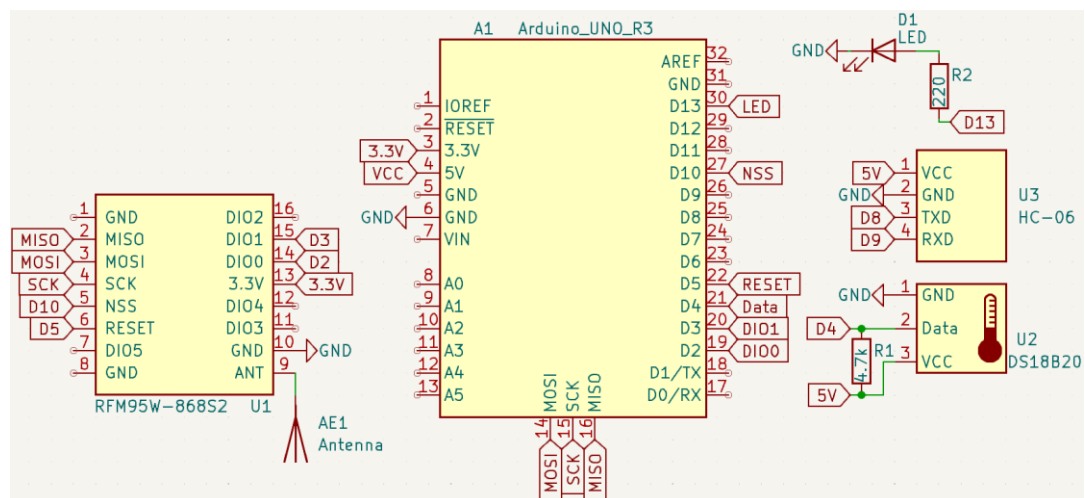


Figura 43: Esquemático e ligações do fieldTester.

O *shield* foi desenvolvido através da utilização de uma placa perfurada e ligações por cabo soldadas no lado inferior da placa. Os pinos e os *headers* de ligação ao Arduino foram soldados de forma a cumprir com os espaçamentos da placa de desenvolvimento e a garantir a possibilidade de utilização de todos os pinos do Arduino. O módulo de comunicação LoRa e *Bluetooth* foram soldados diretamente na placa devido à sua dimensão, ao espaçamento entre pinos e eficácia de comunicação, enquanto que o sensor de temperatura se encontra ligado a um *header*, de forma a permitir uma troca do componente mais facilitada. Na Figura 44 é possível observar uma representação física do fieldTester desenvolvido.

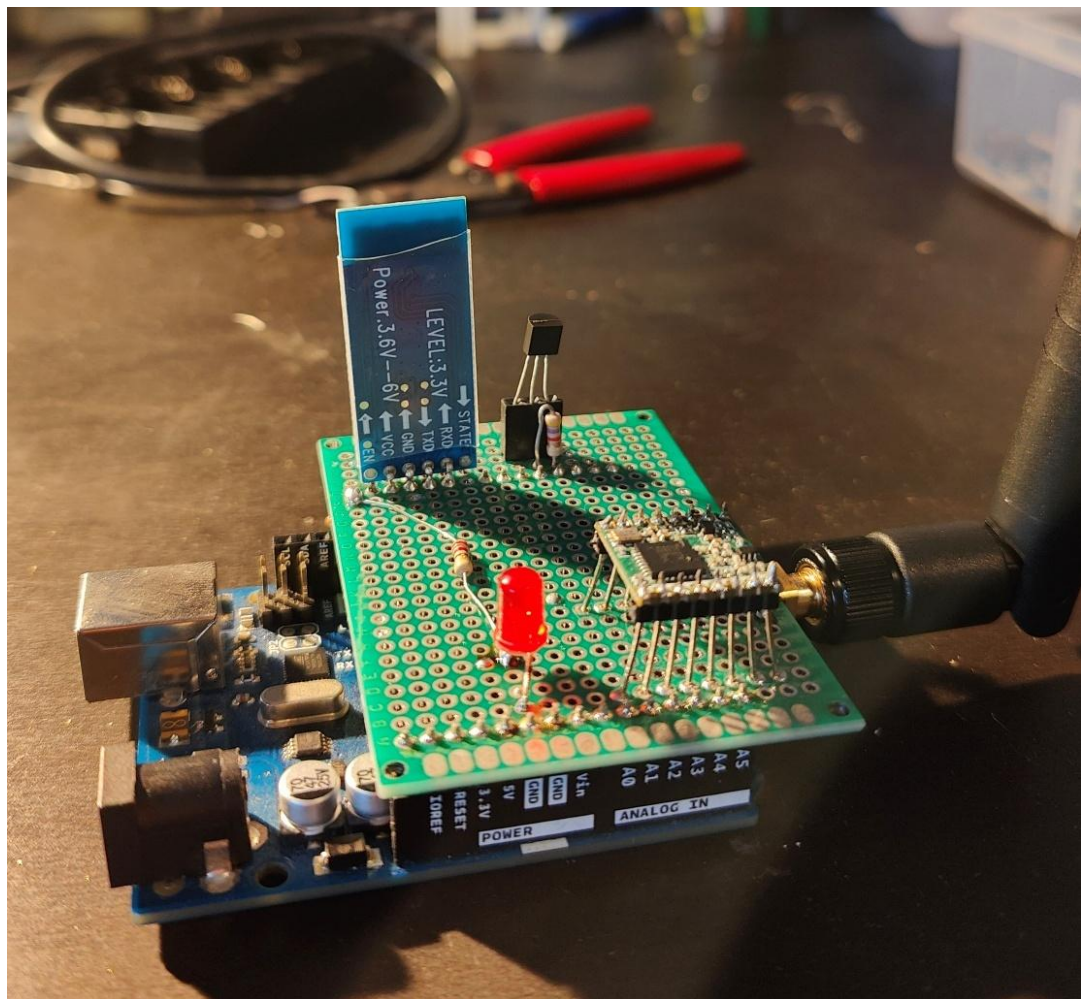


Figura 44: Representação física do fieldTester.

Utilizando o Arduino IDE e com base no código exemplo “ttn-abp” da biblioteca “IBM LMIC framework”, foi desenvolvido o código necessário à integração do dispositivo na plataforma TTSS da TTN através de ativação ABP. De forma a adaptar-se às características do dispositivo e para que a sua leitura permita uma compreensão sucinta dos conceitos base da comunicação LoRa, este código foi simplificado e desenvolvido com o objetivo de integrar facilmente um dispositivo de testes e de comprovar o sucesso da comunicação e da transmissão de dados.

A Figura 45 apresenta o fluxograma do código desenvolvido para o fieldTester que representa o seu comportamento e os diferentes processos executados ao longo do seu funcionamento. A inicialização do *Bluetooth* e do sensor de temperatura é constituída pela inicialização dos objetos referentes a estes, através da definição dos pinos do Arduino onde estes se encontram ligados. A inicialização e configuração da comunicação LoRa é constituída pela atribuição das chaves de ativação ABP geradas pelo servidor de rede, pela definição das variáveis que armazenam o processo paralelo de envio (*sendjob*), a mensagem, o intervalo de tempo entre transmissões e o valor de RSSI e ainda, pela definição dos pinos do Arduino onde o módulo RFM95 se encontra ligado. Um processo paralelo, denominado de “job”, têm um comportamento semelhante a uma *thread*, possuindo uma execução paralela e de segundo plano, sem necessidade de colocar o código associado na função cíclica *loop()*. A função *do_send()* é executada a cada ciclo de transmissão, verificando se existem transmissões ou receções de dados em execução, para transformar os dados (*dataString*) num vetor de caracteres e transmiti-los. A função *onEvent()* verifica se a transmissão foi bem sucedida, aguarda por uma possível receção de dados e apresenta o último valor válido de RSSI, sendo que este é novamente calculado caso exista um *downlink*. Por fim, obtém-se um novo valor de temperatura e “agenda-se” a próxima transmissão de dados, de acordo com o intervalo de tempo definido e com a função *do_send()*. O valor do RSSI é obtido através da função *getRSSI()*, sendo que esta é executada sempre que se detete a receção de uma mensagem. A determinação da intensidade do sinal é baseada nos cálculos fornecidos pelo datasheet do módulo LoRa SX1276 que constitui o módulo RFM95 [104]. O valor de temperatura é obtido através da função *getTemp()*, na qual esse valor é validado e transformado em “*String*”, para que seja posteriormente transmitido por LoRa. A função *setup()* inicializa o *Bluetooth*, o sensor de temperatura e o módulo LoRa e define os parâmetros a utilizar nesta sessão de comunicação LoRa, como as chaves, os canais de comunicação ou o *SF* da janela de receção. A função *loop()* mantém em constante execução o processo paralelo (*sendjob*). Assim, o comportamento do fieldTester pode ser resumido da seguinte forma:

- Caso não existam transmissões ou receções de dados em curso, o dispositivo tenta enviar o valor de temperatura por comunicação LoRa. Caso a comunicação tenha sido bem-sucedida, o dispositivo aguarda por uma eventual receção de dados.
- Caso exista uma receção de dados, o LED indicador é ligado, o RSSI é calculado e a mensagem recebida é apresentada.
- Por fim, apresenta o último valor calculado de RSSI, envia-o por *Bluetooth* e mede um novo valor de temperatura. Após estas operações, o dispositivo aguarda o tempo programado até ao próximo ciclo de comunicação.

O código desenvolvido é apresentado no Anexo 1 desta dissertação.

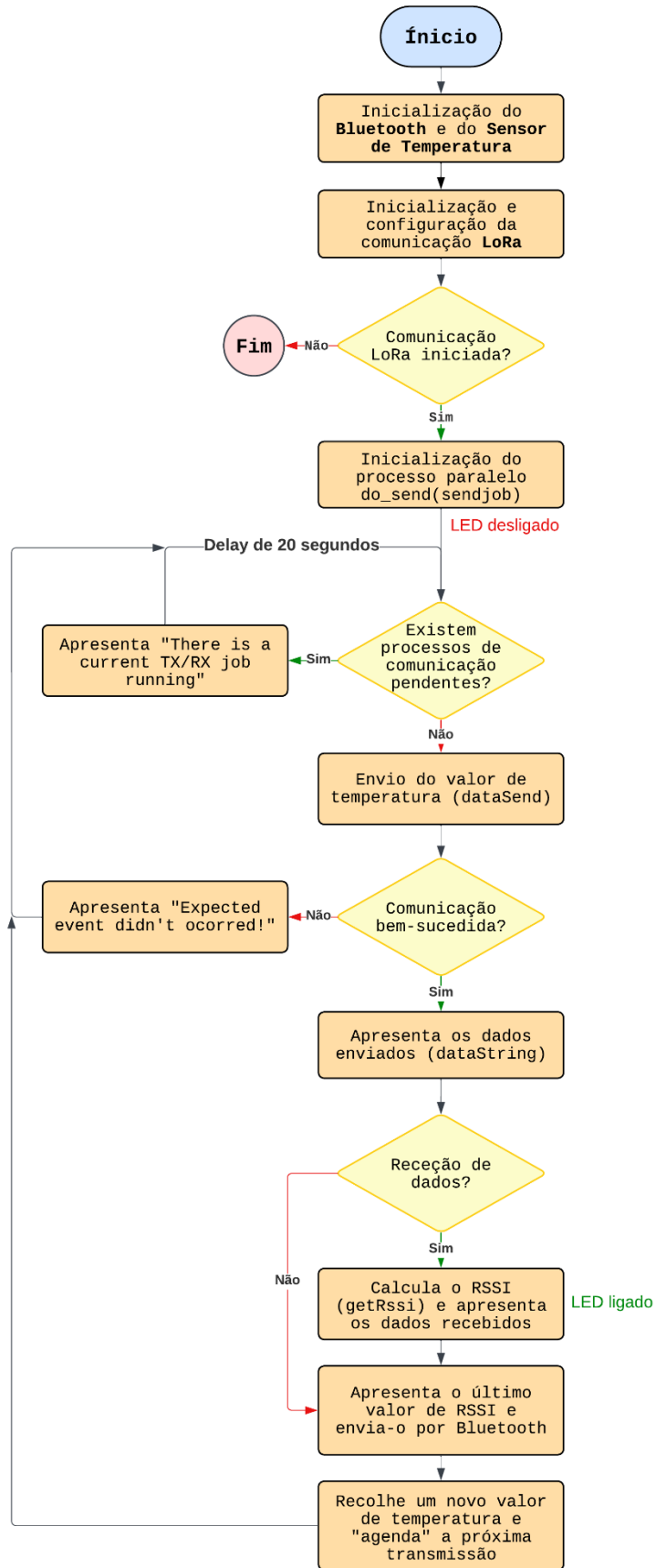


Figura 45: Fluxograma do código desenvolvido para o fieldTester.

6.1.2 Sensor magnético de porta (Dragino LDS02)

O nó sensor Dragino LDS02 é um sensor magnético que verifica a abertura e fecho de portas através da detecção de um íman na sua proximidade. O dispositivo é constituído pela sua componente principal capaz de realizar a comunicação LoRa e algumas tarefas de análise de dados e de estado e ainda pela sua componente de menor dimensão constituída por um íman, de forma a realizar a força magnética necessária à sua detecção. O objetivo principal da aplicação deste dispositivo é a detecção da abertura da porta da “Sala de apoio técnico e logístico aos laboratórios” do DEE-ET e registo das alterações deste estado para monitorização do acesso dos docentes à sala e eventuais acessos indevidos. Esta aplicação permite criar alertas de abertura indevida da porta, monitorizar o estado de abertura/fecho e registar o tempo em que a porta permaneceu em cada estado, entre outros tipos de situações programáveis através da informação do estado da porta, da etiqueta de data/hora de cada alteração de estado, do contador de aberturas da porta, do tempo em que a porta ficou aberta ou de uma mensagem de alarme. Na Tabela 10 é possível observar o formato do *payload* de uma mensagem *uplink* do dispositivo. Esta abrangência de aplicações permite que este dispositivo seja também aplicável numa janela ou em qualquer outra superfície deslocável, na qual se queira monitorizar as alterações de estado ou a entrada/saída de ativos. Este dispositivo utiliza o método de ativação OTAA e possui chaves próprias fornecidas pelo fabricante, facilitando a sua integração em qualquer servidor de rede. Na Figura 46 é possível observar o local de instalação do Dragino LDS02.

Tabela 10: *Payload* de *uplink* do Dragino LDS02 [98]

Tamanho (bytes)	2	1	3	3	1
Item	Status e Bateria	MOD (0x01)	Nº total de eventos	Duração da última abertura (minutos)	Alarme



Figura 46: Local de instalação do Dragino LDS02.

6.2 Gateway LoRa

A *gateway* LoRa da Kerlink foi implementada no telhado do ponto de referência anteriormente mencionado, junto ao bloco F do edifício da Escola Superior de Tecnologia de Setúbal, no qual se encontram as salas e infraestruturas utilizadas pelo ramo de Eletrónica e Telecomunicações do Departamento de Engenharia Eletrotécnica e desta forma, a *gateway* estará aproximadamente localizada no centro do *campus* e o seu acesso fica mais facilitado por parte dos responsáveis pela rede. A *gateway* encontra-se ligada à Internet através de uma rede virtual específica localizada junto ao local de instalação e é possível acedê-la facilmente através do seu IP estático. O acesso à *gateway* pode ser feito através da interface *web* da mesma, onde é possível consultar o estado, reiniciar a *gateway* ou atualizar o seu *firmware* ou ainda através de SSH e da linha de comandos da *gateway* para configurações mais avançadas. Utilizando uma aplicação como o Putty, é possível aceder à *gateway* por SSH e proceder à configuração da mesma, nomeadamente, à associação da *gateway* ao servidor de rede TTSS da TTN e respetiva utilização da chave API criada no servidor de rede. Substituindo “API-KEY” pela chave criada, executa-se o comando de configuração na linha de comandos da *gateway* e esta inicia a comunicação com o servidor de rede, permitindo a sua própria gestão e a gestão dos dados através do mesmo.

Na Figura 47 é possível observar o local de instalação da *gateway*, na Figura 48 encontra-se representada a interface *web* de configuração da *gateway* e consultando o mapa de cobertura da Figura 73 é possível observar a localização específica da *gateway* no mapa do *campus*.



Figura 47: Local de instalação da *gateway*.

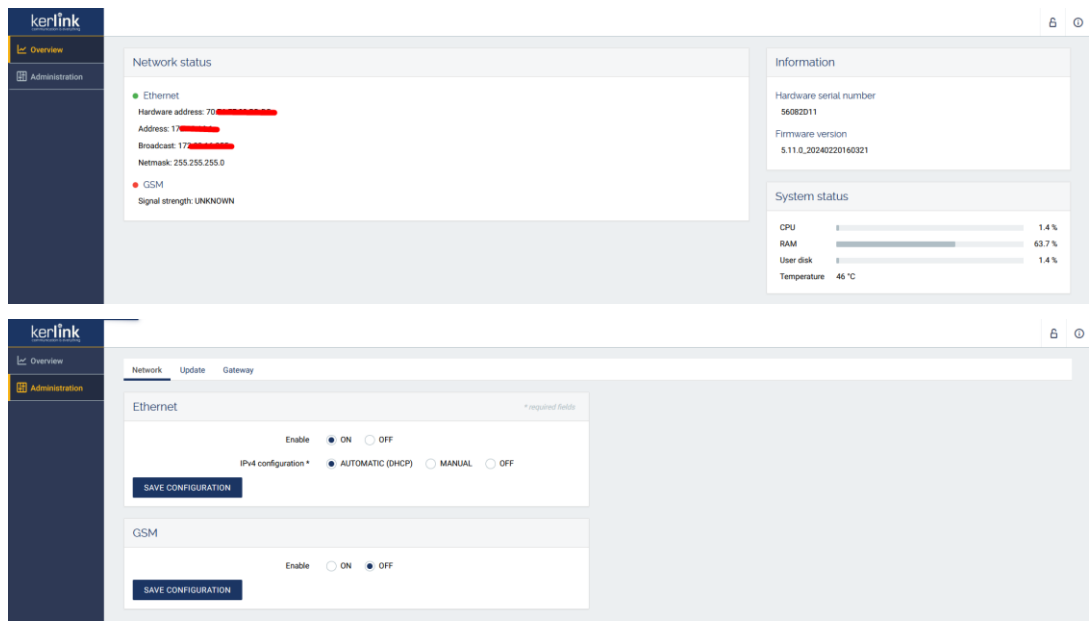


Figura 48: Interface web de configuração da gateway.

6.3 Servidor de rede LoRa

O servidor de rede TTSS da TTN divide a gestão da rede LoRa entre *gateways* e aplicações onde é possível integrar conjuntos de dispositivos com configurações próprias e distintas de outras aplicações e dispositivos. Após a criação da conta na plataforma web do servidor de rede, o utilizador tem acesso às *gateways* adicionadas e às aplicações criadas, sendo que a *dashboard* inicial apresenta de forma acessível as principais entidades criadas (aplicações, *gateways* e nós sensores) e as notificações relacionadas com criações ou alterações nas mesmas, Figura 49.

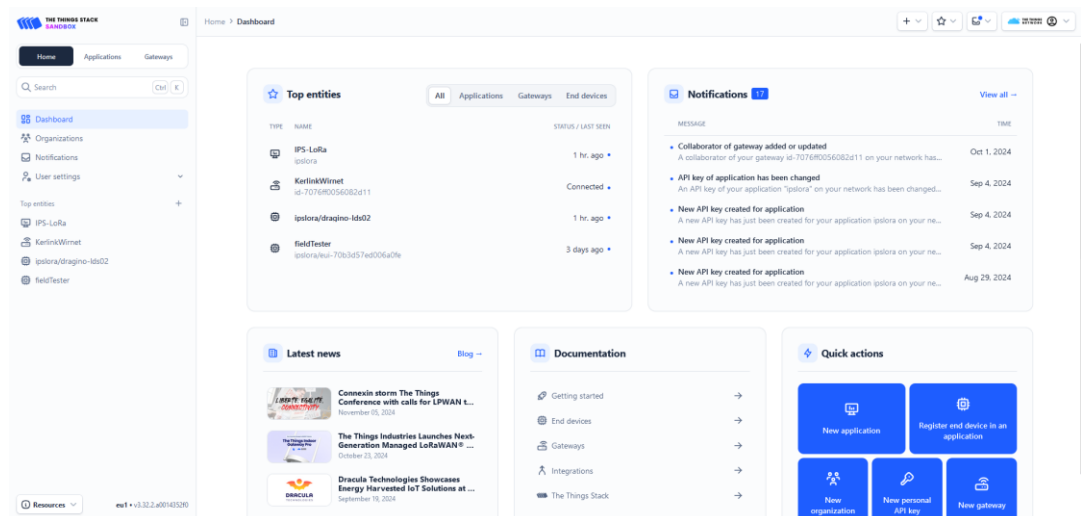


Figura 49: Dashboard inicial - TTSS.

Acedendo ao separador “Gateways”, é possível registar uma *gateway* através do seu identificador EUI (Extended Unique Identifier), associando um identificador próprio para o servidor e a frequência de operação de acordo com a sua localização e fator de dispersão pretendido para a janela de receção de dados RX2, Figura 50.

Register gateway

Register your gateway to enable data traffic between nearby end devices and the network.
Learn more in our guide on [Adding Gateways](#).

Does your gateway have a LoRaWAN® Gateway Identification QR Code? Scan it to speed up onboarding.

[Scan gateway QR code](#)

Gateway EUI [ⓘ]

7E 7C [redacted] [Reset](#)

Gateway ID [ⓘ] *

id-707E [redacted]

Gateway name [ⓘ]

KerlinkWirnet

Frequency plan [ⓘ] *

Europe 863-870 MHz (SF9 for RX2 - recommended) | v

[+ Add frequency plan](#)

Note: most gateways use a single frequency plan. Some 16 and 64 channel gateways however allow setting multiple within the same band.

[Require authenticated connection [ⓘ]](#)

Choose this option eg. if your gateway is powered by [LoRa Basic Station](#)

Share gateway information

Select which information can be seen by other network participants, including [Packet Broker](#)

[Share status within network [ⓘ]](#)

[Share location within network [ⓘ]](#)

[Register gateway](#)

Figura 50: Registo da *gateway* Kerlink Wirnet iStation - TTSS.

Após o registo da *gateway* KerlinkWirnet, é apresentada a sua página e uma visão geral (*overview*) das informações, configurações, estado e localização da *gateway*, Figura 51.

The screenshot displays the 'Gateway overview' page for 'KerlinkWirnet' (ID: 34-707E). The interface is divided into several sections:

- General information:** Gateway ID (34-707E), Gateway EUI (7E 7C [redacted]), Frequency plan (Europe 863-870 MHz (SF9 for RX2 - recommended)), and Created at (May 25, 2024 19:30:03).
- Network settings:** A list of settings with toggle switches: Automatic updates (Disabled), Require authenticated connection (Disabled), Public status (Enabled), Public location (Enabled), Packet Broker forwarding (Enabled), Status location updates (Enabled), and Enforce duty cycle (Enabled).
- Gateway status:** Shows a 30-day uptime of 771.92ms and a roundtrip time of 771.92ms. It also displays connection status for various frequencies (863.0-865.0MHz, 865.0-866.0MHz, 866.0-867.0MHz, 867.0-868.0MHz, 868.0-869.0MHz) with their respective duty cycle utilization percentages (0.00%, 0.26%, 0.00%, 0.00%, 0.00%).
- Location:** A map showing the gateway's location in a city area.

Figura 51: Visão geral da *gateway* KerlinkWirnet - TTSS.

Na aba "Location" é possível configurar a partilha da localização da *gateway* e definir a sua localização de forma manual ou automática, consoante a existência de um módulo de geolocalização na *gateway*. Esta partilha de informação sobre a localização da *gateway* é útil para a componente comunitária do servidor de rede e permite que outros utilizadores saibam onde existem *gateways* LoRa, Figura 52.

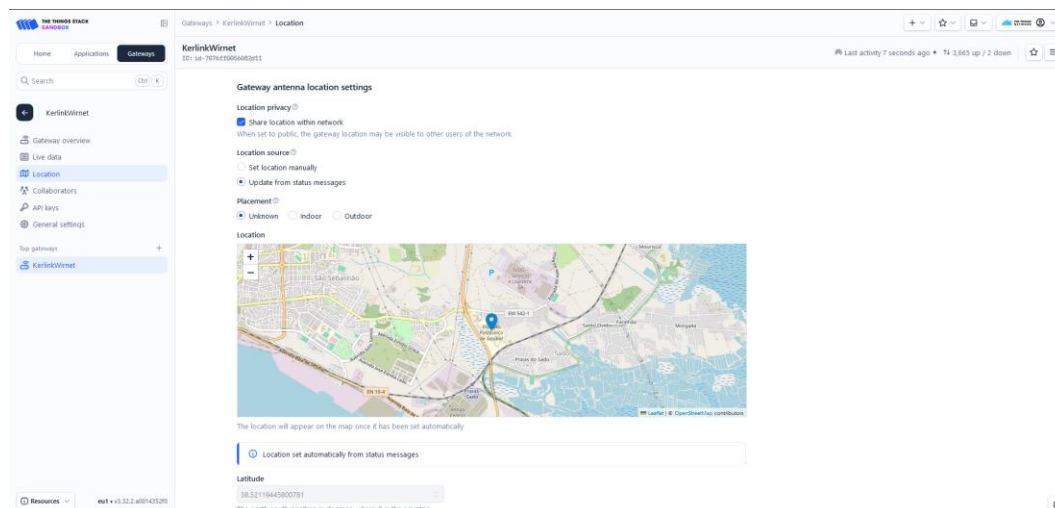


Figura 52: "Location" da *gateway* KerlinkWimnet - TTSS.

Na aba "API keys" da *gateway* é possível criar chaves de acesso à entidade *gateway*. Neste caso, foi apenas criada a chave API utilizada no comando de configuração da *gateway* na secção 6.2, mas é também possível criar acessos a esta entidade de forma a que a *gateway* física seja também gerida por outros servidores, Figura 53.

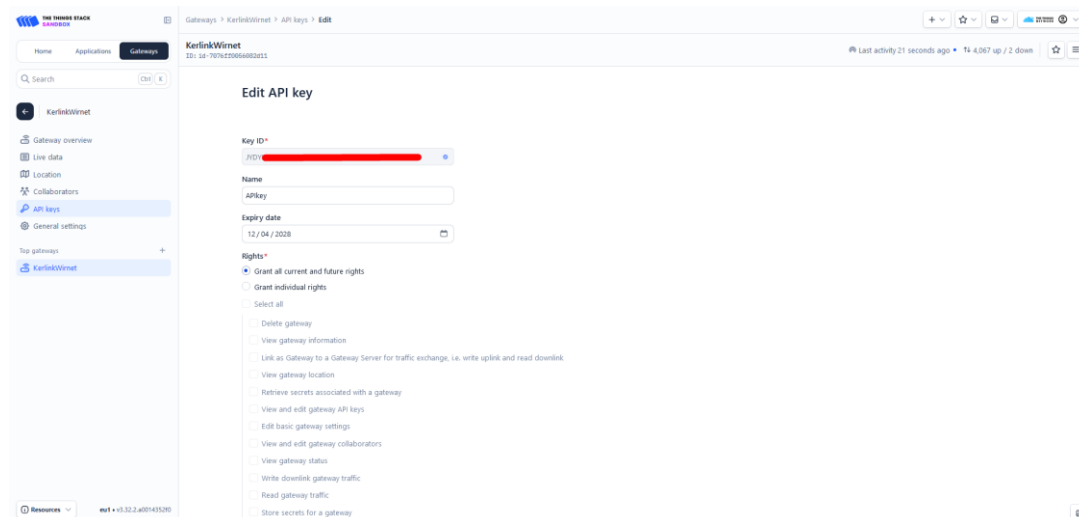


Figura 53: "API keys" da *gateway* KerlinkWimnet - TTSS.

Acedendo ao separador “Applications”, é possível registar aplicações que permitem integrar os nós sensores, gerir as suas configurações e dados e utilizar as muitas opções existentes de integração com outros servidores e serviços externos de componente aplicacional. Após o registo da aplicação IPS-LoRa, foi possível aceder à sua página e obter uma visão geral dos dispositivos integrados, do último *payload* recebido e da localização dos dispositivos, Figura 30.

Na aba “End devices” da aplicação é possível integrar os nós sensores através da importação de um ficheiro com os dados dos diferentes dispositivos, através do repositório de dispositivos da TTN ou através da introdução manual dos parâmetros necessários à ativação do dispositivo. Na Figura 54 é possível observar a integração do nó sensor Dragino LDS02 através do repositório do servidor de rede.

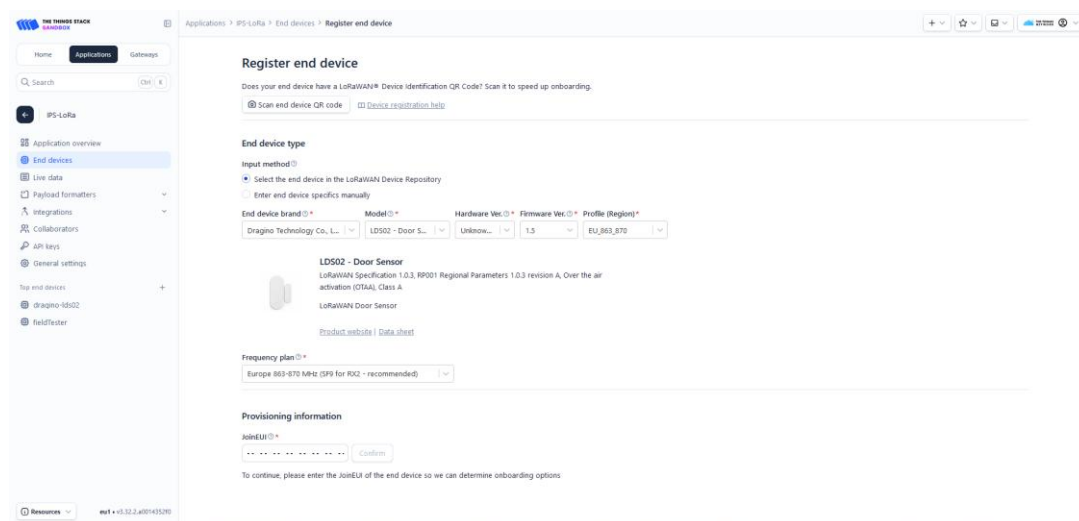


Figura 54: Integração do Dragino LDS02 - TTSS.

Visto que o fieldTester é um dispositivo programável, torna-se necessário que a sua integração seja manual, para que todas as suas especificações sejam registadas e para que as suas chaves de autenticação sejam geradas. O fieldTester foi configurado para utilizar a versão do protocolo LoRaWAN 1.0.3 e classe de comunicação A, sendo ativado pelo método ABP. As chaves de autenticação foram geradas pelo servidor de rede e atribuídas ao dispositivo, tendo sido estas utilizadas na programação do dispositivo, Figura 55.

Register end device

Does your end device have a LoRaWAN® Device Identification QR Code? Scan it to speed up onboarding.

[Scan end device QR code](#) [Device registration help](#)

End device type

Input method [Ⓞ]

Select the end device in the LoRaWAN Device Repository

Enter end device specifics manually

Frequency plan [Ⓞ] *

Europe 863-870 MHz (SF9 for RX2 - recommended)

LoRaWAN version [Ⓞ] *

LoRaWAN Specification 1.0.3

Regional Parameters version [Ⓞ] *

RP001 Regional Parameters 1.0.3 revision A

[Show advanced activation, LoRaWAN class and cluster settings](#)

Activation mode [Ⓞ]

Over the air activation (OTAA)

Activation by personalization (ABP)

Define multicast group (ABP & Multicast)

Additional LoRaWAN class capabilities [Ⓞ]

None (class A only)

Network defaults [Ⓞ]

Use network's default MAC settings

Cluster settings [Ⓞ]

Skip registration on Join Server

Provisioning information

DevEUI [Ⓞ]

... .. 8/50 used

Device address [Ⓞ] *

... ..

AppSKey [Ⓞ] *

... ..

NwKsKey [Ⓞ] *

... ..

End device ID [Ⓞ] *

my-new-device

After registration

View registered end device

Register another end device of this type

Figura 55: Integração manual do fieldTester - TTSS.

Na aba “Live data” da aplicação é possível observar em tempo real os dados recolhidos e comunicados pelos nós sensores integrados na aplicação. Nesta página da aplicação é possível observar as informações de cada dispositivo e o *payload* por estes transmitido, na formatação que lhe for atribuída, Figura 56.

TIME	ENTITY ID	TYPE	DATA PREVIEW
14:36:18	eui-78b3d57e088a08e	Forward uplink data message	DevAddr: 26 88 D6 F8 Payload: { text: "22.31" } 32 32 2E 33 31 FPort: 1 Data rate: SF98M25 SNR: 8.8 RSSI: -74
14:36:18	dragino-lds02	Forward uplink data message	DevAddr: 26 88 51 68 Payload: { ALARM: 0, BAT_V: 3.138, DOOR_OPEN_STATUS: 1, DOOR_OPEN_TIMES: 126, LAST_DOOR_OPEN_DURATION: 0, MOD: 1 } 32 32 2E 33 35 FPort: 1 Data rate: SF98M25 SNR: 18 RSSI: -67
14:36:55	eui-78b3d57e088a08e	Forward uplink data message	DevAddr: 26 88 D6 F8 Payload: { text: "22.25" } 32 32 2E 32 35 FPort: 1 Data rate: SF98M25 SNR: 18 RSSI: -67
14:36:51	dragino-lds02	Forward uplink data message	DevAddr: 26 88 51 68 Payload: { ALARM: 0, BAT_V: 3.138, DOOR_OPEN_STATUS: 1, DOOR_OPEN_TIMES: 126, LAST_DOOR_OPEN_DURATION: 0, MOD: 1 } 32 32 2E 33 35 FPort: 1 Data rate: SF98M25 SNR: 18 RSSI: -67
14:36:32	eui-78b3d57e088a08e	Forward uplink data message	DevAddr: 26 88 D6 F8 Payload: { text: "22.19" } 32 32 2E 31 39 FPort: 1 Data rate: SF98M25 SNR: 12.5 RSSI: -74
14:36:14	dragino-lds02	Forward uplink data message	DevAddr: 26 88 51 68 Payload: { ALARM: 0, BAT_V: 3.138, DOOR_OPEN_STATUS: 0, DOOR_OPEN_TIMES: 124, LAST_DOOR_OPEN_DURATION: 0, MOD: 1 } 32 32 2E 31 39 FPort: 1 Data rate: SF98M25 SNR: 12.5 RSSI: -74
14:36:09	eui-78b3d57e088a08e	Forward uplink data message	DevAddr: 26 88 D6 F8 Payload: { text: "22.19" } 32 32 2E 31 39 FPort: 1 Data rate: SF98M25 SNR: 9.2 RSSI: -70
14:36:05	dragino-lds02	Forward uplink data message	DevAddr: 26 88 51 68 Payload: { ALARM: 0, BAT_V: 3.126, DOOR_OPEN_STATUS: 0, DOOR_OPEN_TIMES: 123, LAST_DOOR_OPEN_DURATION: 14, MOD: 1 } 32 32 2E 31 39 FPort: 1 Data rate: SF98M25 SNR: 11.8 RSSI: -70
14:34:56	dragino-lds02	Forward uplink data message	DevAddr: 26 88 51 68 Payload: { ALARM: 0, BAT_V: 3.12, DOOR_OPEN_STATUS: 0, DOOR_OPEN_TIMES: 123, LAST_DOOR_OPEN_DURATION: 14, MOD: 1 } 32 32 2E 31 39 FPort: 1 Data rate: SF98M25 SNR: 11.8 RSSI: -70
14:34:47	eui-78b3d57e088a08e	Forward uplink data message	DevAddr: 26 88 D6 F8 Payload: { text: "22.19" } 32 32 2E 31 39 FPort: 1 Data rate: SF98M25 SNR: 13.2 RSSI: -78
14:34:25	eui-78b3d57e088a08e	Forward uplink data message	DevAddr: 26 88 D6 F8 Payload: { text: "22.13" } 32 32 2E 31 33 FPort: 1 Data rate: SF98M25 SNR: 11.8 RSSI: -76
14:34:02	eui-78b3d57e088a08e	Forward uplink data message	DevAddr: 26 88 D6 F8 Payload: { text: "22.19" } 32 32 2E 31 39 FPort: 1 Data rate: SF98M25 SNR: 12.2 RSSI: -76
14:33:40	eui-78b3d57e088a08e	Forward uplink data message	DevAddr: 26 88 D6 F8 Payload: { text: "Hello" } 48 68 6C 6C 6F FPort: 1 Data rate: SF98M25 SNR: 12.8 RSSI: -72

Figura 56: "Live data" da aplicação IPS-LoRa - TTSS.

Na aba “Payload formatters” da aplicação é possível definir uma formatação para todos os dispositivos da aplicação, dividida por comunicações *uplink* ou *downlink*. Estas formatações podem ser realizadas através de *scripts* personalizados JavaScript ou outros serviços de formatação oferecidos pelo servidor, Figura 57. É possível atribuir ainda uma formatação específica a cada dispositivo dentro da página dos mesmos, como pode ser observado na Figura 64 e Figura 65.

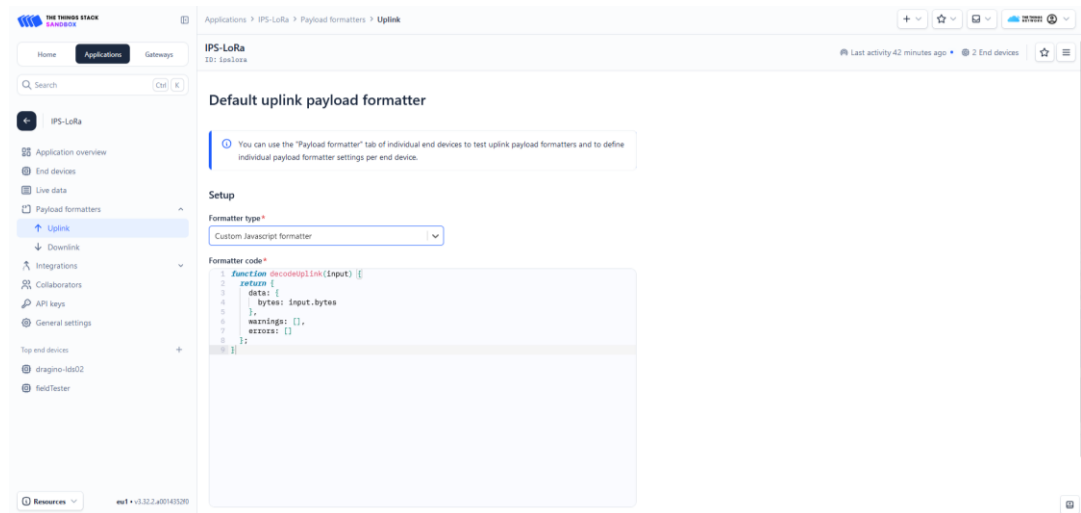


Figura 57: "Payload formatters" da aplicação IPS-LoRa - TTSS.

Na aba "Integrations" da aplicação é possível observar as informações e configurações relativas à integração de dados através de MQTT, ao armazenamento destes em bases de dados, às integrações diretas com os serviços AWS IoT, Azure IoT e LoRa Cloud e ainda a integração com outros serviços e servidores aplicativos através de *webhooks*. A integração com o MQTT permite uma comunicação dos dados mais facilitada e divergente através da ligação ao MQTT *broker* da TTN, com o qual a maioria dos serviços e dos dispositivos possui compatibilidade. Ainda assim, existe também a possibilidade de criar uma ligação em formato *webhook*, que comunica os dados sempre que exista a receção destes por parte da *gateway*, sendo este mais utilizado na comunicação entre aplicações e servidores em formato *web*.

Na aplicação IPS-LoRa foram criadas duas *webhooks* de forma a estabelecer ligação com o servidor aplicativo TagoIO e com o serviço TTN Mapper, Figura 58. O TTN Mapper é uma ferramenta *online* que acede aos dados da aplicação através da *webhook* e permite obter mapas de cobertura e informações mais interativas sobre os dispositivos da rede LoRa. Criando esta *webhook* e partilhando a localização da *gateway*, é possível pesquisar de forma específica por estes dispositivos e observar as suas diversas ligações e informações num mapa interativo [115]. Na Figura 59 é possível observar o registo de ligações da *gateway*, sendo que a precisão dos dados depende bastante da informação que o utilizador disponibiliza e da forma como utiliza os seus nós sensores.

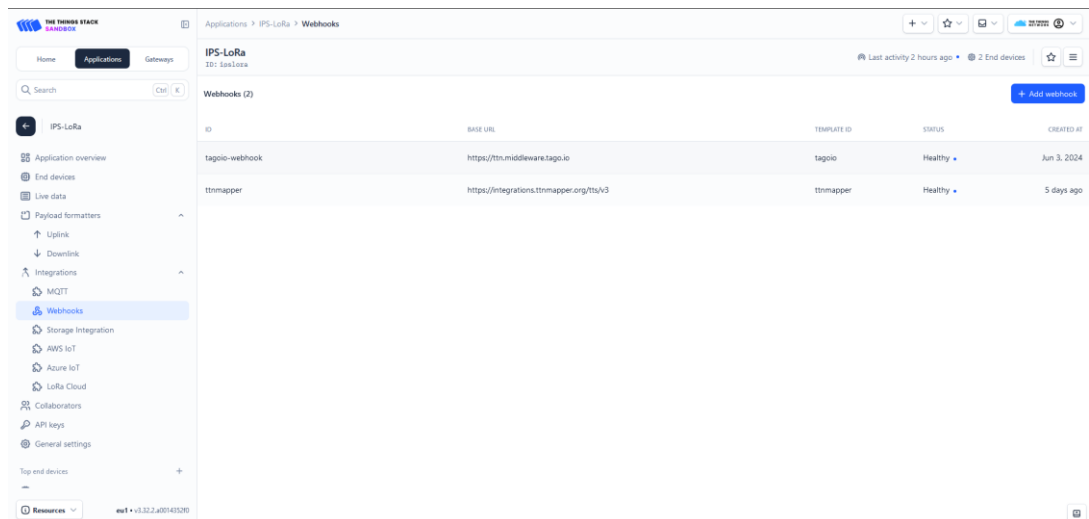


Figura 58: "Webhooks" da aplicação IPS-LoRa - TTSS.

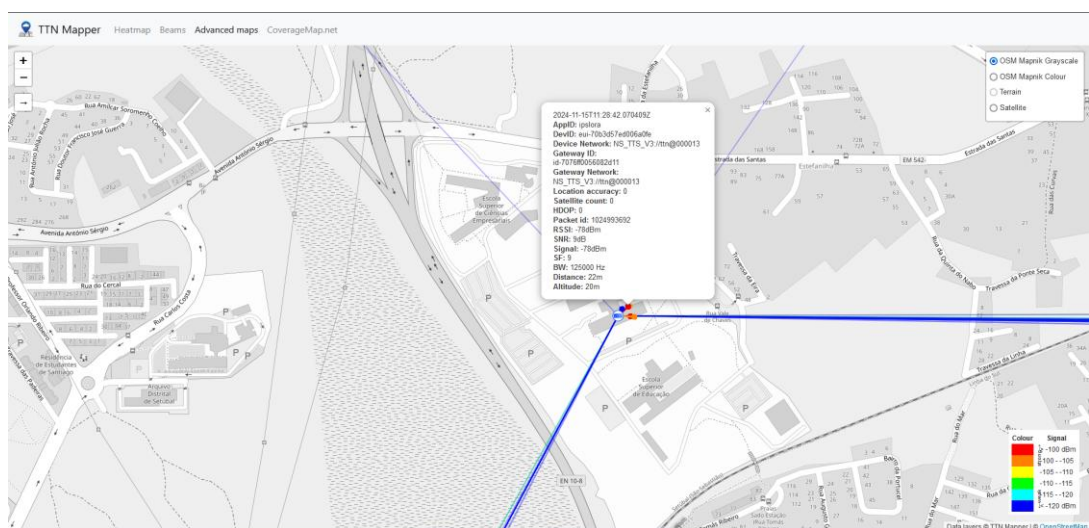


Figura 59: Ligações da gateway KerlinkWirnet - TTN Mapper.

Na aba "API keys" da aplicação é possível criar chaves de acesso à aplicação e aos nós sensores e permitir a gestão de comunicações *uplink* e *downlink* através de outros servidores ou aplicações. Na aplicação IPS-LoRa, foi gerada uma chave API para comunicação MQTT através da aba "Integrations" e outra chave API especificamente para permitir realizar comunicações *downlink* através do servidor aplicativo TagoIO, Figura 60.

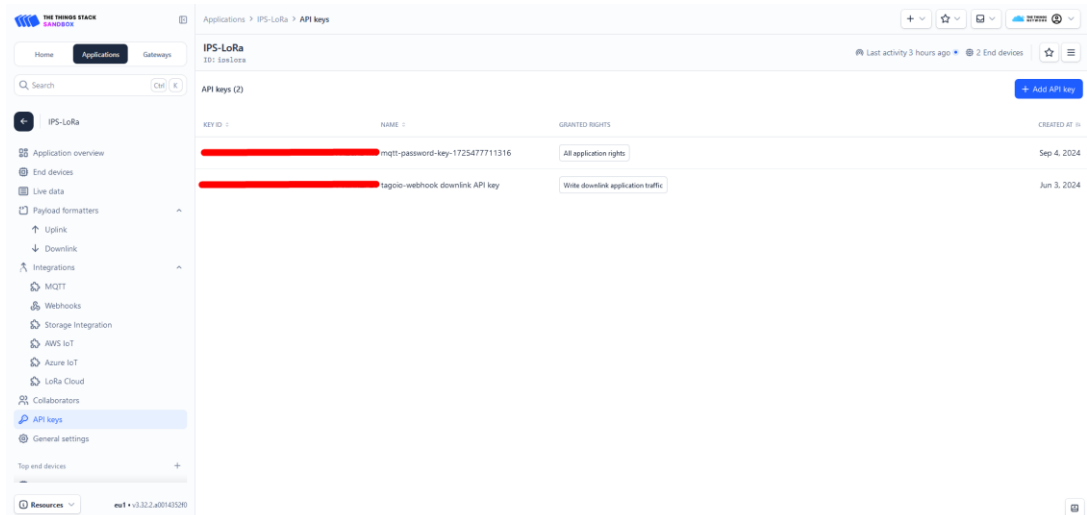


Figura 60: "API keys" da aplicação IPS-LoRa - TTSS.

Na página de cada dispositivo é possível obter uma visão geral do dispositivo integrado e aceder a outras funcionalidades como dados em tempo real (Live data), teste de comunicação *uplink* e *downlink*, geolocalização, formatação de *payload* e configurações relacionadas com o protocolo LoRaWAN, Figura 61.

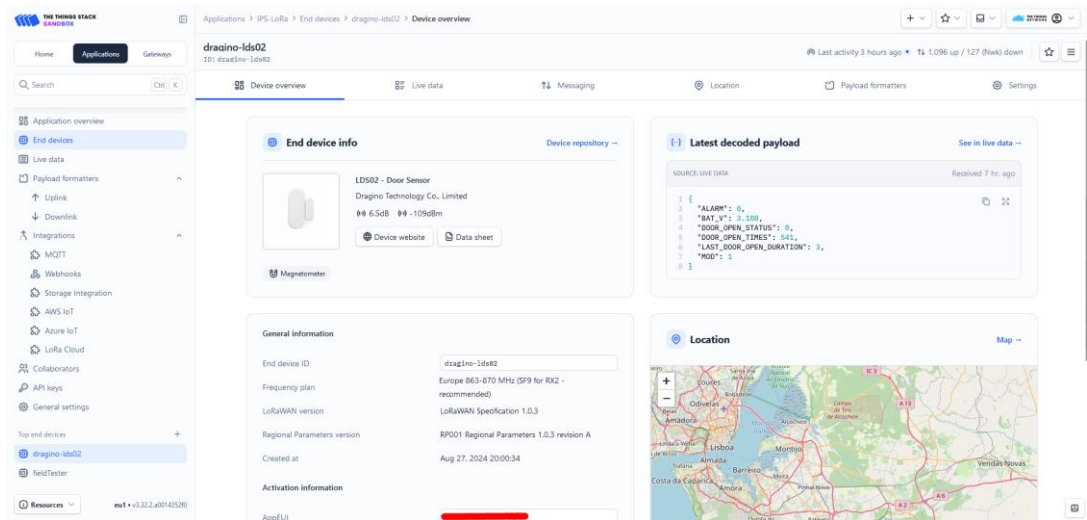


Figura 61: "Device overview" do nó sensor Dragino LDS02 - TTSS.

Na aba "Messaging" dos nós sensores é possível realizar uma comunicação *downlink*, enviando *payloads* para o nó sensor, sendo que esta funcionalidade será utilizada para obter o valor de RSSI no nó sensor e é através desta que a comunicação é executada pelo servidor aplicacional TagIO, Figura 62. Nesta aba é ainda possível simular uma comunicação *uplink* com o mesmo objetivo de realizar testes de comunicação.

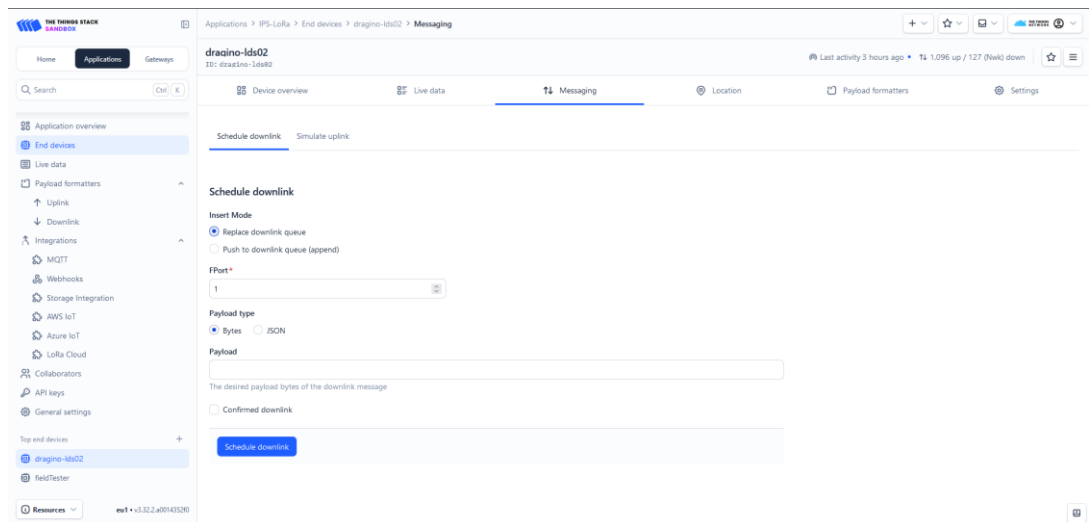


Figura 62: Comunicação *downlink* em "Messaging" do nó sensor Dragino LDS02 - TTSS.

Na aba "Location" dos nós sensores é possível atribuir manualmente coordenadas de localização aos nós sensores, caso estas não sejam detetadas no *payload* enviado pelos nós sensores. Esta partilha de informação sobre a localização dos nós sensores é útil para a componente comunitária do servidor de rede e permite que outros utilizadores saibam onde existem nós sensores LoRa e a qualidade da sua comunicação, Figura 63.

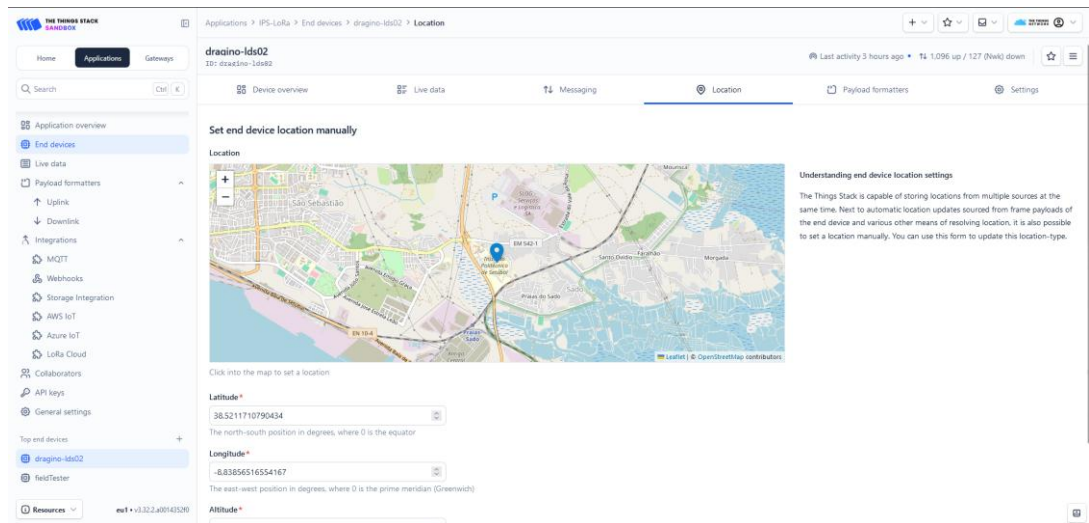


Figura 63: "Location" do nó sensor Dragino LDS02 - TTSS.

Na aba "Payload formatters" dos nós sensores é possível definir formatações específicas para cada dispositivo, divididas por comunicações *uplink* ou *downlink*. A formatação do *payload* do nó sensor Dragino LDS02 é realizado através de um *script* próprio fornecido pela fabricante e presente no repositório do servidor de rede, Figura 64. A formatação do *payload* do fieldTester foi desenvolvida através de um *script* JavaScript e de acordo com o formato da mensagem programada, Figura 65.

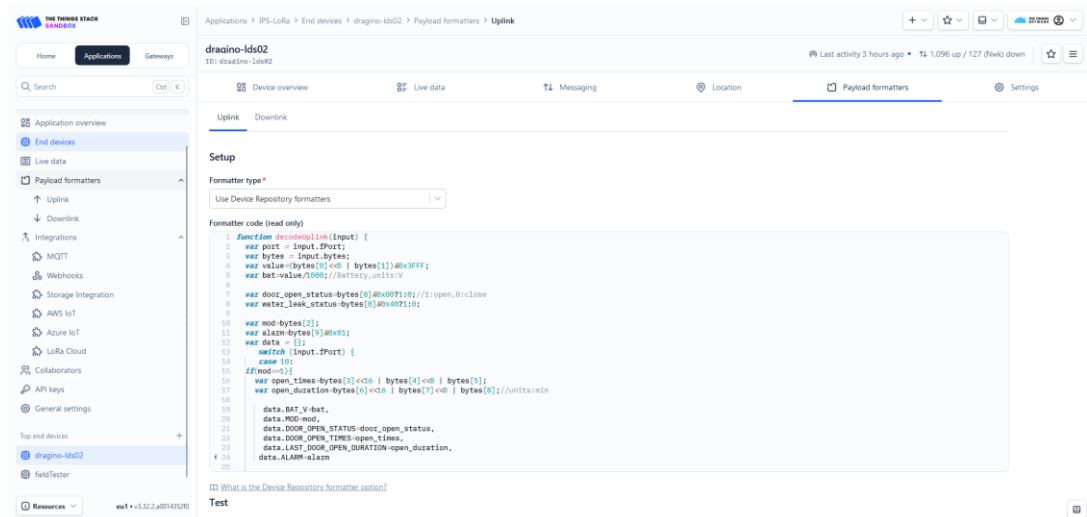


Figura 64: Formatação do *payload* do nó sensor Dragino LDS02 - TTSS.

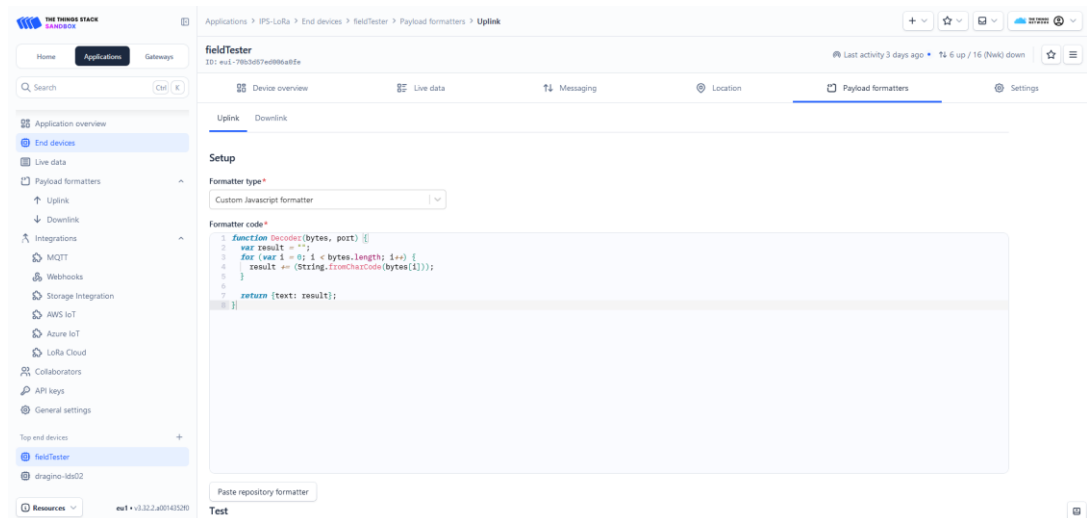


Figura 65: Formatação do *payload* em "Payload formatters" do fieldTester - TTSS.

Na aba "Settings" dos nós sensores é possível modificar as opções selecionadas na integração do dispositivo e ainda alterar configurações importantes relacionadas com a comunicação LoRa, como as frequências a utilizar ou a ativação do ADR. Nas configurações do fieldTester, o ADR foi desativado para garantir a consistência dos testes e a utilização dos mesmos parâmetros em diferentes locais de teste e foram ainda atribuídas frequências de operação programadas também no dispositivo para limitar o número de frequências utilizadas, o que permite garantir maior eficiência e estabilidade na comunicação, Figura 66.

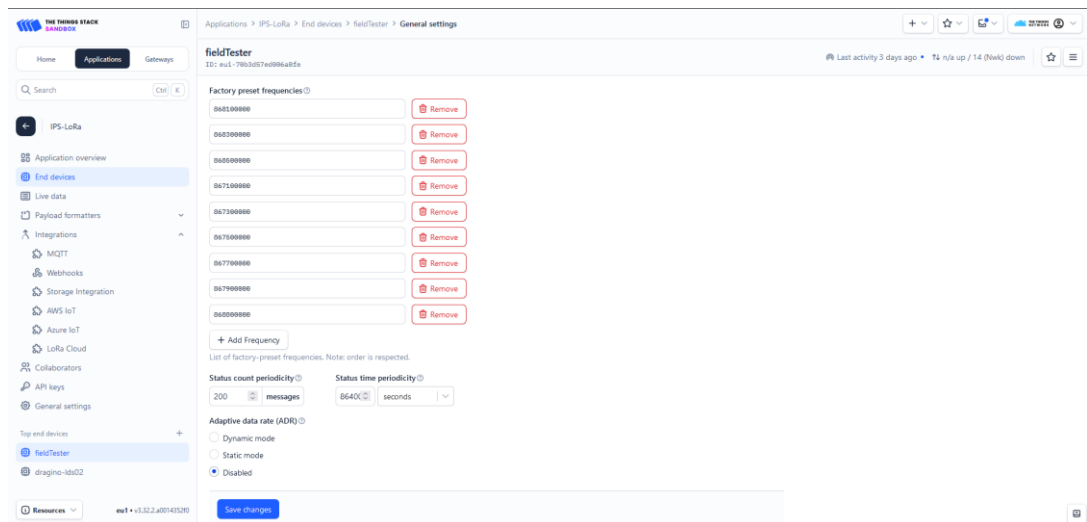


Figura 66: "Settings" do fieldTester - TTSS.

6.4 Servidor aplicativo IoT

Um servidor aplicativo IoT é tipicamente compatível com diversas formas de comunicação para que exista uma integração correta dos dados, seja qual for a arquitetura e tecnologias utilizadas na rede IoT. O TagoIO possui mais de 500 conectores, o que significa que pode processar e receber dados de inúmeros dispositivos e formas de comunicação mais generalizadas. Neste caso, os dispositivos encontram-se integrados num servidor de rede da TTN e o TagoIO possui integração direta com estes, pelo que basta criar um *token* de autorização e implementar a *webhook* necessária no servidor de rede, para que os dispositivos criados no TagoIO tenham acesso aos dados correspondentes da TTSS. Este *token* de autorização foi previamente criado e a *webhook* já se encontra implementada anteriormente, como é possível observar na Figura 58. Caso estes servidores não possuíssem uma integração direta, a comunicação entre eles poderia facilmente ser realizada através de MQTT ou HTTP.

Na opção "Devices" do menu do TagoIO é possível criar novos dispositivos, selecionando o tipo de ligação desejada. Para adicionar os dispositivos anteriormente integrados no TTSS, foi necessário selecionar o conector "Custom TTI/TTN" e atribuir o EUI referente a cada nó sensor integrado.

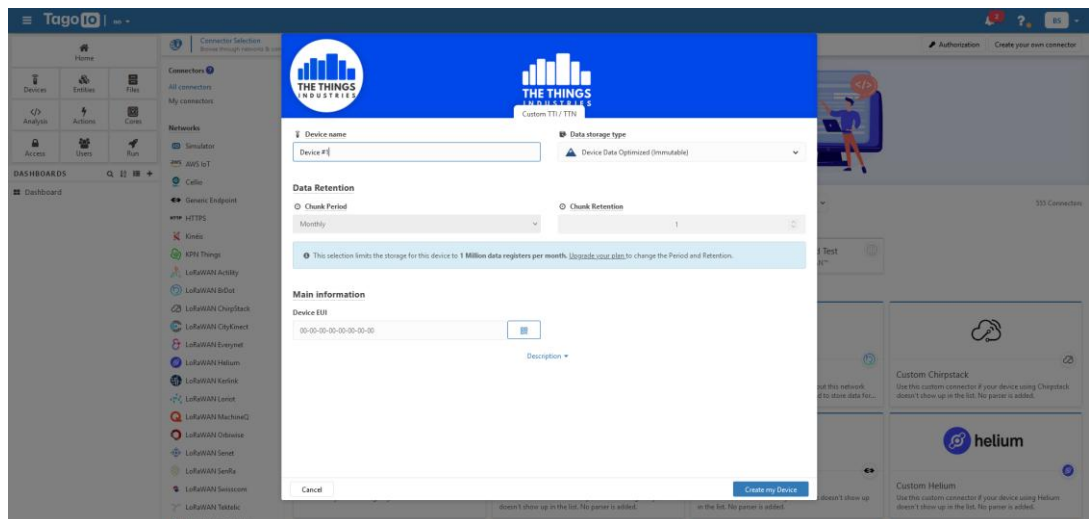


Figura 67: Criação de um dispositivo - TagoIO.

Após a criação de um dispositivo, é possível observar os dados recebidos agrupados em variáveis retiradas do *payload*, gerir esses dados, simular uma nova entrada de valores no armazenamento simplificado do dispositivo, formatar o *payload*, inspecionar os dados em tempo real ou alterar parâmetros do dispositivo, como as chaves de autenticação. Na Figura 68 é possível observar a organização das variáveis do *payload*, sendo que cada uma destas pode ser utilizada nos *scripts* de automação ou apresentada de diferentes formas numa *dashboard*.

ID	Variable	Value	Group	Time
4811f	frm_payload	0:0001000220000800 (string)	1731954470329	hh 5 hours
4811a	snr	8 (number)	1731954470329	hh 5 hours
4811d	ssi	-103 (number)	1731954470329	hh 5 hours
4811c	gateway_eui	7078FF005602D11 (string)	1731954470329	hh 5 hours
4811b	font	1006 (number)	1731954470329	hh 5 hours
4811a	fpout	10 (number)	1731954470329	hh 5 hours
48119	application_id	ipidore (string)	1731954470329	hh 5 hours
48118	device_id	dragino-ls02 (string)	1731954470329	hh 5 hours
48117	time	2024-11-18T18:27:50.003Z (string)	1731954470329	hh 5 hours
48116	timestamp	247809823 (number)	1731954470329	hh 5 hours
48115	frequency	86730000 (string)	1731954470329	hh 5 hours
48114	lora_coding_rate	4/5 (string)	1731954470329	hh 5 hours
48113	lora_spreading_factor	7 (number)	1731954470329	hh 5 hours
48112	lora_bandwidth	125000 (number)	1731954470329	hh 5 hours
48111	mod	1 (number)	1731954470329	hh 5 hours
48110	last_door_open_duration	137 (number)	1731954470329	hh 5 hours

Figura 68: “Data” do nó sensor Dragino LDS02 - TagoIO.

Após a criação dos dois dispositivos, foi também criada uma *dashboard* para apresentar os valores de temperatura do sensor presente no fieldTester, o estado de abertura/fecho da porta e um botão para realizar uma comunicação *downlink* para o fieldTester. Através da capacidade de armazenamento do TagoIO é também possível adicionar tabelas e gráficos de histórico de valores, tendo sido adicionado duas tabelas para os valores dos sensores e um gráfico de variação de temperatura medida pelo sensor.

Tendo em conta o contexto e o propósito deste projeto, foi apenas utilizado o armazenamento simplificado de cada dispositivo que, na versão gratuita do TagoIO, permite armazenar até 800 000 registos e realizar até 1 000 000 de comunicações por mês. A *dashboard* criada está presente na Figura 69.



Figura 69: *Dashboard* IPS-LoRa - TagoIO.

O botão de *downlink* presente na *dashboard* tem como objetivo o envio de uma mensagem simples para o fieldTester, de forma a que este possa calcular o RSSI dessa comunicação. O comportamento do botão é programado através de um *script* desenvolvido na opção “Analysis”, utilizando um *template* fornecido pelo TagoIO e a chave API criada especificamente para comunicação *downlink* no servidor de rede. O TagoIO permite ainda que os botões e *scripts* criados sejam implementados individualmente em formato *web* noutro tipo de serviços, dispositivos ou aplicações, o que permitiu realizar a comunicação *downlink* de forma muito mais facilitada através da aplicação móvel criada. A mensagem “313233” é enviada em formato de *bytes*, sendo depois decodificada para formato *String* e representando a mensagem “123”. Na Figura 70 é possível observar algumas configurações possíveis do botão e o URL que permite implementar o botão na aplicação móvel.

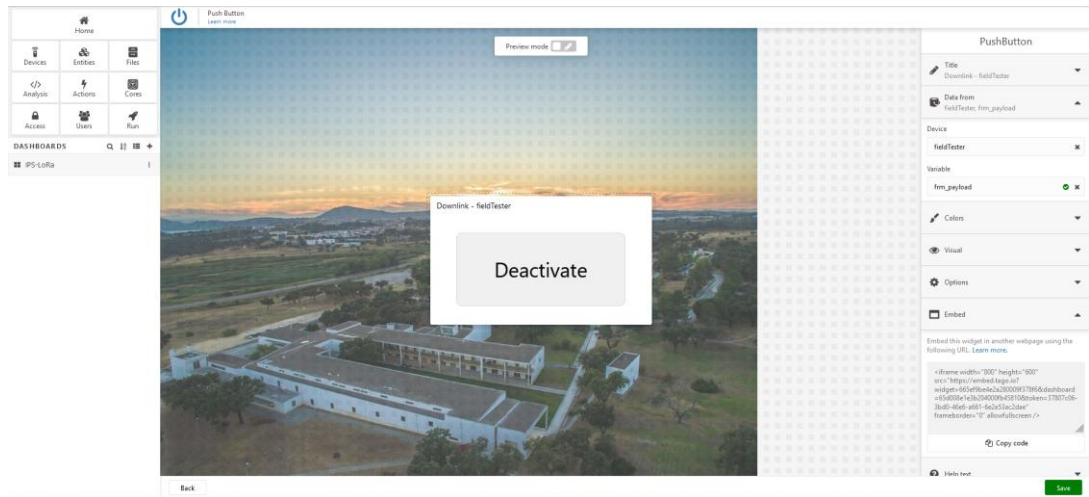


Figura 70: Botão “Downlink” da *dashboard* IPS-LoRa – TagoIO.

7. Testes de cobertura

Os testes de cobertura têm como principal objetivo a avaliação do alcance e da qualidade de comunicação numa determinada área e utilizando determinados dispositivos e tecnologias. No contexto desta dissertação, os testes de cobertura permitiram determinar a distância máxima a que os dispositivos conseguem comunicar entre si, verificar o impacto que os obstáculos possuem no alcance e na qualidade do sinal e ainda identificar “zonas sombra” dentro do raio de alcance máximo da rede. Estes testes contribuem ainda para a determinação da melhor localização da *gateway* e das melhores configurações a utilizar consoante o local de implementação. Os mapas de cobertura resultantes destes testes apresentam o valor de RSSI obtido no dispositivo de testes para cada local de teste. Neste capítulo será apresentado o desenvolvimento da aplicação móvel utilizada para recolher os valores de RSSI e coordenadas de geolocalização, o processo de recolha de dados, os mapas de cobertura criados e uma análise aos resultados obtidos.

7.1 Medição da intensidade de sinal recebido

Visto que um dos objetivos desta dissertação é o estudo da tecnologia de comunicação LoRa e do seu potencial, torna-se necessário analisar a qualidade da receção de dados e a que distância se encontra do emissor, principalmente dos nós sensores. A análise da receção de dados nos nós sensores torna-se ainda mais importante, visto que a *gateway* possui tipicamente melhor capacidade de comunicação e que as limitações serão normalmente devido à menor capacidade dos nós sensores, dado que operam com potências inferiores e possuem antena com menor ganho.

Esta análise pode ser realizada através da criação de mapas de cobertura em que se associa cada valor de RSSI a coordenadas de geolocalização. De forma a simplificar ao máximo o processo de associação e a limitar a complexidade do dispositivo de testes, foi criada uma aplicação móvel que recebe os valores de RSSI calculados pelo dispositivo através de *Bluetooth* e os associa às coordenadas de geolocalização do dispositivo móvel, armazenando estes valores num ficheiro .csv em que cada entrada tem o seguinte formato: [latitude, longitude, RSSI]. O processo de cálculo do RSSI implica que exista uma comunicação *downlink* para avaliar a qualidade do sinal recebido no nó sensor, sendo que este processo é assegurado através do botão e do *script* desenvolvido no servidor aplicacional TagoIO. Utilizando a ferramenta *online* ApplInventor para desenvolver a aplicação móvel, foi possível integrar esse botão como uma componente *web* na aplicação e realizar a comunicação *downlink* através da aplicação desenvolvida, sem que seja necessário aceder ao TagoIO.

Na Figura 71 é possível observar o fluxograma da aplicação móvel e analisar o processo de receção e armazenamento de dados e ainda os diferentes processos paralelos que ela executa relacionados com a comunicação *downlink* e com a determinação das coordenadas. O código de blocos desenvolvido encontra-se apresentado na íntegra no anexo 2 do Anexo desta dissertação.

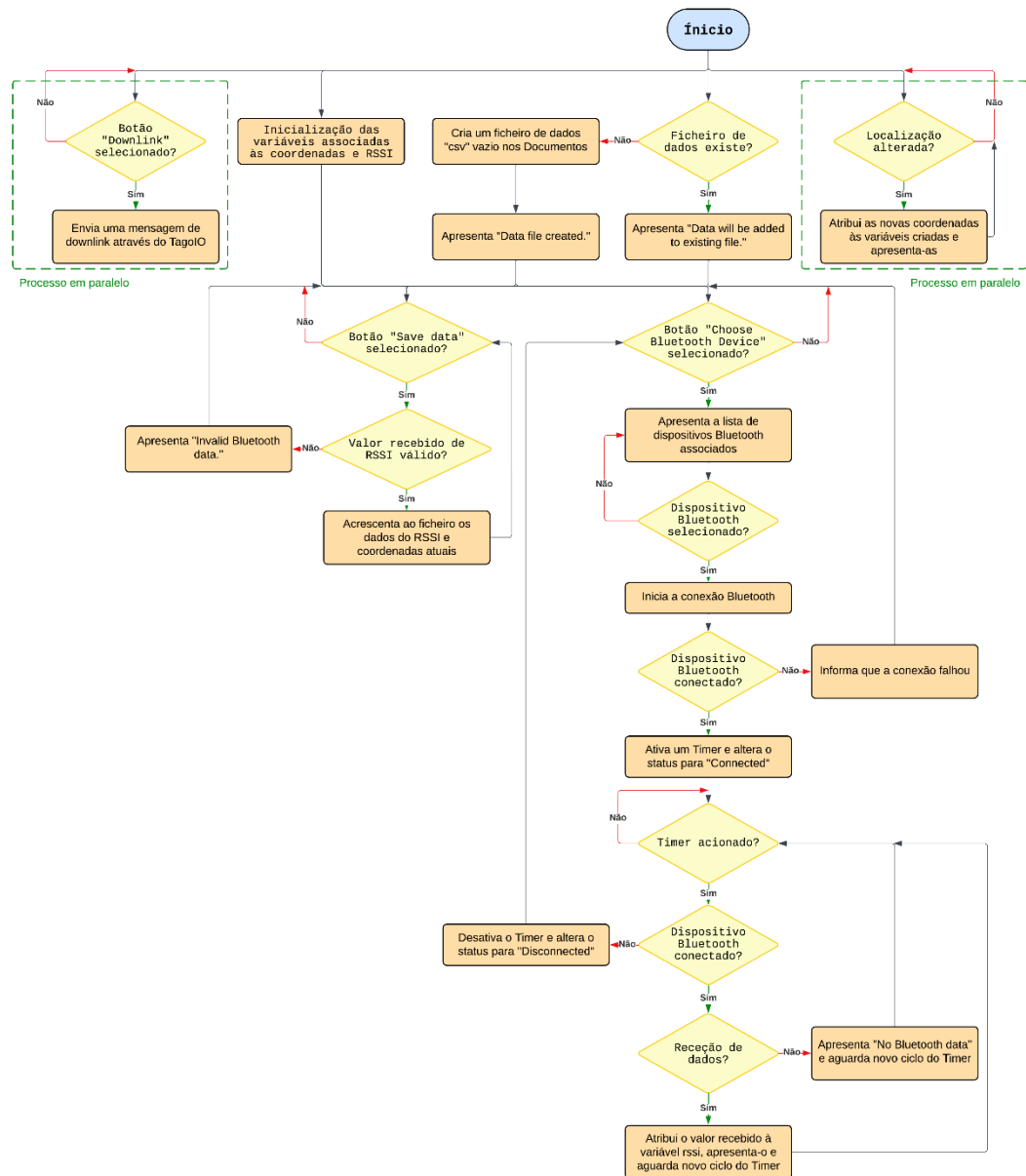


Figura 71: Fluxograma da aplicação móvel (RSSI_GPS).

7.2 Recolha de dados

O processo de recolha de dados inicia-se pela deslocação ao local de teste e pelo posicionamento do fieldTester num local preferencialmente desobstruído. Após ligar o dispositivo, é possível selecioná-lo na lista de dispositivos associados na aplicação móvel e a comunicação *Bluetooth* é iniciada. Pressionando o botão de *downlink* e aguardando que a comunicação seja realizada, o valor de RSSI é apresentado e é possível armazená-lo através do botão “Save GPS and RSSI data”. Em paralelo, ocorre a recolha e apresentação das coordenadas do dispositivo móvel sempre que se detete uma alteração das mesmas. Ao ligar o dispositivo, este comunica o valor do sensor de temperatura a cada 20 segundos, o que significa que a janela de receção de dados também só se abre a cada 20 segundos e que é necessário aguardar esse tempo para que a comunicação *downlink* seja realizada. No entanto, o botão da aplicação que executa essa comunicação pode ser pressionado a qualquer altura e esta comunicação fica pendente de execução no servidor de rede até que exista um *uplink* do dispositivo. A receção de dados por *Bluetooth* é também realizada a cada 20 segundos, informando o utilizador da existência ou não existência de dados a receber através da apresentação do valor de RSSI ou da informação “No Bluetooth Data” respetivamente. Na Figura 72 é possível observar a única página da aplicação móvel RSSI_GPS e onde é apresentada toda a informação.

Alguns testes intermédios determinaram que a receção dos dados fosse realizada utilizando apenas um fator de dispersão SF9 e desativando a funcionalidade de ADR por oferecer maior estabilidade geral na comunicação. Ainda assim, sendo um fator de dispersão relativamente equilibrado entre alcance e capacidade de transmissão, é também um dos que mais se utiliza e que abrange um maior número de implementações.

A recolha de dados foi efetuada em locais de interesse que possam integrar dispositivos IoT, como escolas, infraestruturas públicas, parques ou zonas industriais e ainda em locais que sejam relevantes para o estudo do alcance de comunicação. Esta recolha foca-se principalmente na área geográfica a Oeste do *campus* de Setúbal do IPS por esta possuir uma geografia mais diversificada e permitir obter resultados menos constantes e mais interessantes no contexto de estudo da tecnologia de comunicação, enquanto que a área geográfica a Este do *campus* é maioritariamente plana e pouco densa a nível de infraestruturas e obstáculos.

RSSI-GPS App

Choose the Bluetooth device from which you want to receive the RSSI data:

Choose Bluetooth Device

Status: Connected

Downlink button

Activate

GPS

38.50368

-8.92574

RSSI

-129

Press the button below to save GPS and RSSI data.

Save GPS and RSSI data

Figura 72: Aplicação móvel RSSI_GPS.

7.3 Mapa de cobertura

Os mapas de cobertura foram desenvolvidos de forma manual através da ferramenta *online* de *design* Canva e dos mapas da ferramenta *online* Google Maps, observando o ficheiro exportado da aplicação e colocando pontos em cada um dos locais de teste, com atribuição de um valor de RSSI e uma cor a cada um destes, de acordo com o intervalo em que esse valor se situe. Os resultados dos testes realizados nos locais de teste escolhidos para o *campus* de Setúbal do IPS e para a zona de Setúbal encontram-se representados na Tabela 11 e respetivamente, juntamente com a distância horizontal (sem ter em conta a altitude dos pontos) de cada um à *gateway* e o valor de RSSI obtido. Os mapas de cobertura desenvolvidos encontram-se representados na Figura 73 e Figura 74.

Tabela 11: Valores de RSSI para os locais de teste do *campus*

Campus de Setúbal do Instituto Politécnico de Setúbal		
Local	Distância à <i>gateway</i> (m)	Valor de RSSI (dBm)
Locais Interiores		
Sala de apoio técnico e logístico aos laboratórios do DEE-ET	40	- 89
Corredor central do Piso 2 do bloco E da ESTS	43	- 97
Refeitório da ESTS	130	- 104
Locais Exteriores		
Entrada da ESTS	67	- 92
Estacionamento junto à entrada da ESTS	100	- 85
Entrada da ESE	157	- 87
Estacionamento da ESE	164	- 94
Estacionamento exterior junto ao <i>campus</i>	172	- 91
Recinto de atividades académicas	226	- 96
Entrada pedonal Norte do <i>campus</i>	230	- 89
Estacionamento traseiro da ESCE	238	- 102
Estacionamento do pavilhão desportivo	272	- 90
Entrada da ESCE	278	- 105

Zona fitness exterior	290	- 103
Entrada pedonal Noroeste do <i>campus</i>	424	- 110

Tabela 12: Valores de RSSI para os locais de teste da zona de Setúbal e área circundante

Zona de Setúbal e área circundante			
Local	Altitude (m)	Distância à gateway (km)	Valor de RSSI (dBm)
Escola Profissional de Setúbal	16	0,73	- 92
Escola Secundária Dom Manuel Martins	23	1,55	- 110
Terminal Portuário SAPEC	4	2,19	- 118
Zona Comercial de Monte Belo	44	3,04	- 120
Doca do Comércio de Setúbal	6	3,99	- 125
Central Fotovoltaica de Algeruz	6	4,03	- 128
Estação Ferroviária de Setúbal	20	4,20	- 122
Escola Básica Barbosa du Bocage	9	4,69	- 126
Estação Ferroviária de Palmela	45	6,16	- 121
Miradouro do Castelo de Palmela	219	7,15	- 106
Praia da Rasca	7	7,94	- 119
Golf Palmela Village	89	10,88	- 120
Capela das Necessidades	161	12,72	- 120
Zona Industrial de Azeitão	99	13,99	- 126
Convento de Nossa Senhora da Arrábida	236	14,69	-129
Vinícolas de Picheleiros	142	14,74	- 122
Brejos de Azeitão	63	16,23	- 128
Covas de Coina	68	17,5	-121



Figura 73: Mapa de cobertura do *campus* de Setúbal do IPS.

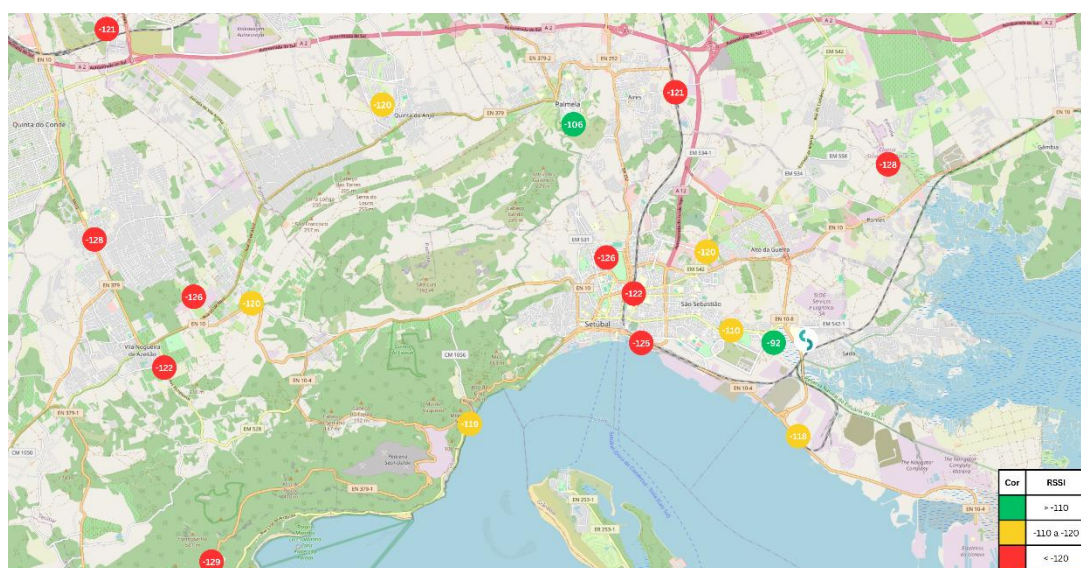


Figura 74: Mapa de cobertura da zona de Setúbal e área circundante.

7.4 Análise de resultados

Em primeira análise é possível concluir que a rede LoRa abrange o *campus* de forma total e que em qualquer local do mesmo será possível implementar dispositivos LoRa e enriquecer a rede IoT implementada. Em relação à zona de Setúbal e área circundante, a cobertura é mais dispersa e bastante afetada pela geografia do local. A zona de Azeitão e o percurso representado pela Estrada Nacional 10 são fortemente afetados pelas suas características montanhosas, como pode ser observado pelos valores de RSSI obtidos. Os testes realizados permitiram verificar uma maior dificuldade na obtenção de valores de RSSI ou a ausência de comunicação em locais imediatamente a norte da Serra da Arrábida ou de baixa altitude ao redor desta. Esta condição pode ser verificada pelos valores semelhantes de RSSI entre os locais “Brejos de Azeitão” e “Zona Industrial de Azeitão”, sendo que estes se encontram a uma altitude semelhante, enquanto que o primeiro local se encontra mais distante. Esta comparação demonstra uma menor capacidade do sinal em ultrapassar o obstáculo quando este se encontra mais próximo do dispositivo e reflete o impacto do efeito sombra dos obstáculos na propagação de sinal.

No local “Praia da Rasca” foi obtido um valor de RSSI semelhante ao do local “Capela das Necessidades” apesar deste último se encontrar mais distante, o que se deve à diferença de altitude e à maior presença de componentes atenuadores entre a “Praia da Rasca” e a *gateway*. Esta comparação demonstra que a presença de obstáculos pode afetar de diferentes formas a transmissão do sinal e que são um fator de enorme importância no planeamento e na implementação de uma rede LoRa ou outras redes que utilizem tecnologias de comunicação de longo alcance.

O alcance e a qualidade da comunicação dependem bastante da densidade de obstáculos do local, especialmente em ambiente urbano, mas dependem também de igual forma da sua altitude, como pode ser observado através da comparação de zonas menos densas como o “Zona Comercial de Monte Belo” ou o “Terminal Portuário SAPEC”. A disparidade de valores obtidos de RSSI explica-se facilmente pela diferença de 38 metros de altitude, o que beneficia o primeiro local, por este se encontrar a uma altitude semelhante à da *gateway*.

Os valores de RSSI obtidos encontram-se dentro do intervalo de valores esperados quando analisada a fórmula de cálculo deste valor e tendo em consideração todas as perdas relacionadas com a propagação de sinal e com a cablagem. Esta análise permite determinar o impacto do ambiente de propagação e utilizá-lo para calcular o valor teórico de RSSI num ponto de geografia semelhante. Através da equação 6 do *Link Budget* e desprezando as perdas da cablagem ($L_{Rx[dB]} = L_{Tx[dB]} = 0$), é possível obter o valor da variável n do fator de atenuação do local “Capela das Necessidades”, o qual se encontra à distância de 12,7 km da *gateway* e onde se obteve um valor de RSSI de -120 dBm.

$$P_{Rx[dBm]} = P_{Tx[dBm]} + G_{Rx[dBi]} + G_{Tx[dBi]} - L_{0[dB]}$$

$$P_{Rx[dBm]} = -120 = 20 + 3 + 3 - (10 * n * \log(12700) + 20 \log(868) - 27,55)$$

$$146 = (41,038n + 58,77 - 27,55)$$

$$n = \frac{114,78}{41,038} = 2,797 \text{ (valor de "n" para ambiente de propagação de "Capela das Necessidades")}$$

Obtido o valor de n para o local “Capela das Necessidades” é possível verificar de forma teórica o valor de RSSI do local “Vinícolas de Picheleiros” a 14,7 km de distância da *gateway*. O valor teórico de RSSI obtido através do cálculo seguinte é semelhante ao valor obtido em contexto real (-122 dBm), no entanto, pode concluir-se que a atenuação geral das perdas deverá ser ligeiramente maior no local “Vinícolas de Picheleiros”.

$$P_{Rx[dBm]} = 20 + 3 + 3 - (10 * 2,797 * \log(14700) + 20 \log(868) - 27,55)$$

$$P_{Rx[dBm]} = -121.78 \text{ dBm (valor teórico de RSSI para "Vinícolas de Picheleiros")}$$

No interior do *campus* e para locais de teste muito próximos são obtidos valores de n superiores devido a uma maior dificuldade de transposição de obstáculos e ocorrência de reflexões por parte do sinal. Calculando o valor de n para o local “Entrada pedonal Norte do *campus*”, o qual se encontra à distância de 230 m da *gateway* e onde se obteve um valor de RSSI de -89 dBm, obtém-se um valor muito superior aos calculados anteriormente.

$$P_{Rx[dBm]} = -89 = 20 + 3 + 3 - (10 * n * \log(230) + 20 \log(868) - 27,55)$$

$$n = \frac{8,378}{2,362} = 3,547 \text{ (valor de "n" para ambiente de propagação de "Entrada pedonal Norte do campus")}$$

Verificou-se ainda que a maior distância de comunicação obtida nos testes foi de 17,5 km nas “Covas de Coima”, tendo sido obtido um valor de RSSI de -121 dBm, o que sugere que o alcance máximo da comunicação entre o fieldTester e a *gateway* possa ser ainda maior, Figura 75. Calculando o valor de n para este local, obteve-se um valor de $n = 2,729$, contribuindo para a conclusão de que o alcance da comunicação LoRa poderá ser maior.

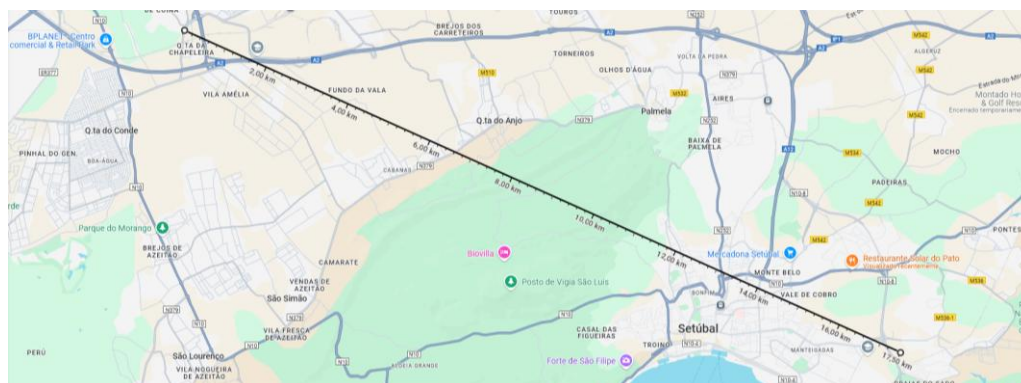


Figura 75: Distância máxima de comunicação obtida nos testes de cobertura.

$$P_{Rx[dBm]} = -121 = 20 + 3 + 3 - (10 * n * \log(17500) + 20 \log(868) - 27,55)$$

$$n = \frac{11,578}{4,243} = 2,729 \text{ (valor de "n" para ambiente de propagação de "Covas de Coina")}$$

Executando um processo automático de determinação de alcance máximo de acordo com os valores utilizados nos cálculos anteriores que descrevem a capacidade de comunicação do fieldTester e de acordo com um valor de n definido, foi possível obter gráficos que demonstram a variação da intensidade de sinal em função da distância e assim, obter o alcance máximo para os valores utilizados. O valor da distância no qual o valor de intensidade de sinal recebido é igual à sensibilidade do fieldTester representa o alcance máximo para as condições apresentadas. A Figura 76 representa o alcance máximo obtido para um valor de n igual ao do local "Capela das Necessidades", sendo que se o ambiente de propagação se mantivesse constante, seria possível comunicar a 47,39 km de distância. Assumindo um valor de n igual ao do local "Covas de Coina", seria possível comunicar a uma distância ainda maior de 61,98 km, como se pode observar pela Figura 77. Assumindo um valor de $n = 2$ para um cenário de propagação em espaço livre, o valor de distância máxima aumenta para os 3459,24 km, o que representa um cenário demasiado irrealista, ultrapassando até o recorde mundial obtido em Portugal de 1336 km [116].

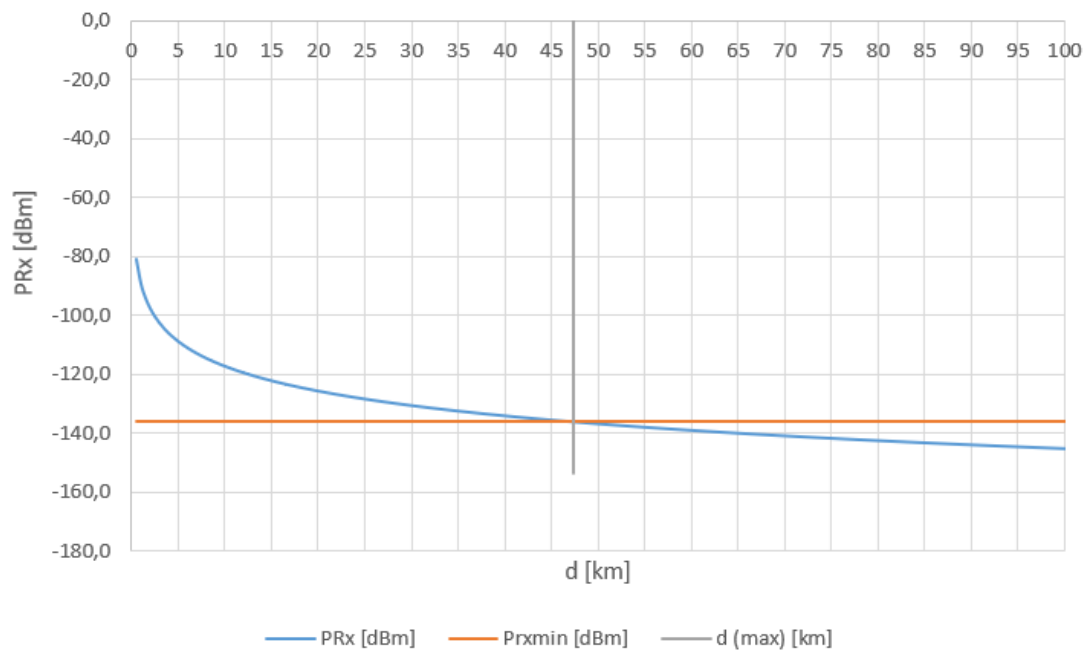


Figura 76: Alcance máximo de 47,39 km para $n = 2,797$.

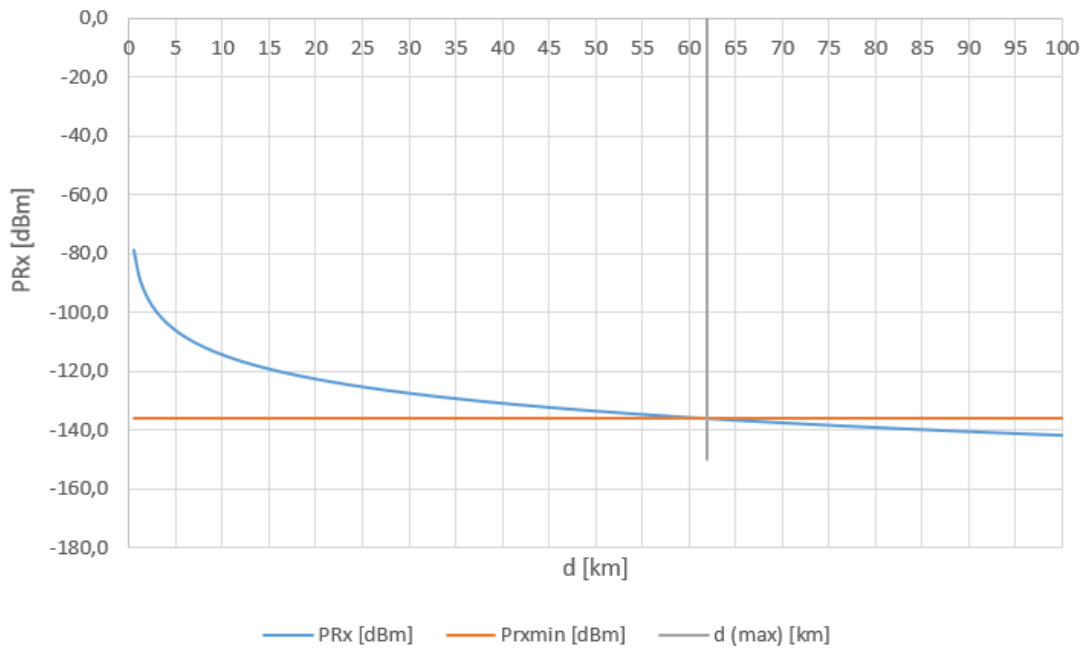


Figura 77: Alcance máximo de 61,98 km para $n = 2,729$.

Analisando os resultados obtidos, é possível concluir que nenhum dos valores obtidos de RSSI se aproximou do valor de sensibilidade do módulo de comunicação LoRa, ou seja, do valor máximo que é possível obter deste parâmetro para as configurações utilizadas. Na Figura 78, é possível observar graficamente os resultados obtidos para a zona de Setúbal e área circundante, tendo em linha de comparação a sensibilidade do módulo de comunicação e as configurações de comunicação utilizadas nos testes (SX1276: Sensibilidade de -136 dBm).

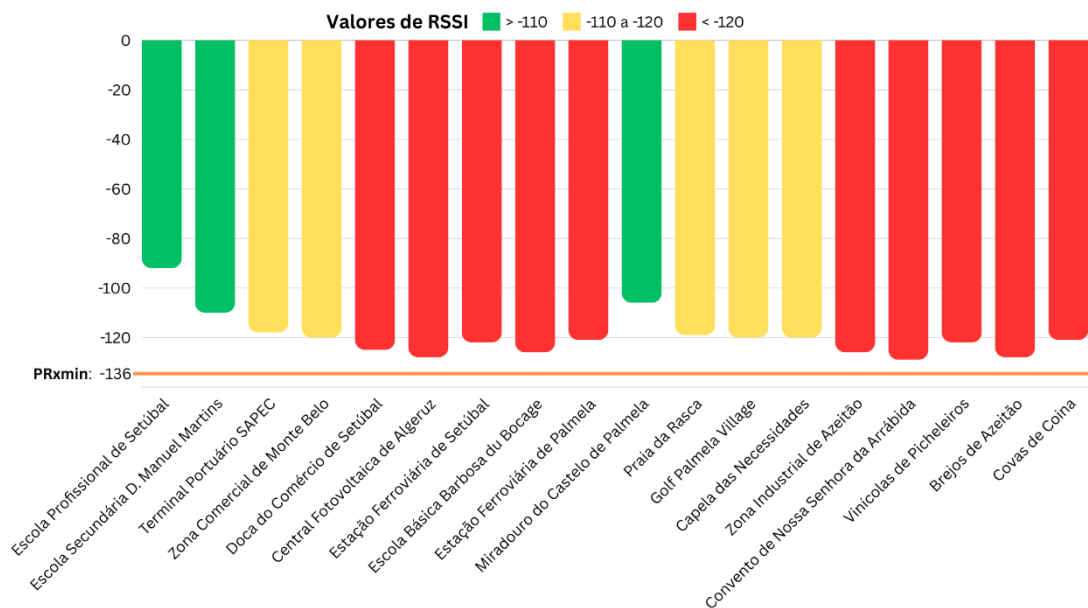


Figura 78: Valores de RSSI para os locais de teste da zona de Setúbal e área circundante.

Estes testes permitiram ainda analisar o comportamento da rede e dos nós sensores e verificar a sua estabilidade e desempenho ao longo do tempo. Na Figura 56 é possível observar a recolha de dados do fieldTester e os diferentes valores de RSSI obtidos pela *gateway* através da receção do sinal do dispositivo de testes, o que permitiu verificar alguma instabilidade no processo de transmissão do mesmo. Esta disparidade pode dever-se ao ruído e às interferências inerentes ao formato de desenvolvimento mais rudimentar do dispositivo, visto que o nó sensor comercial apresenta valores de RSSI bastante mais estáveis.

Conclui-se assim que existem inúmeros parâmetros e fatores a ter em consideração, tanto a nível de *hardware* como de *software* e *firmware*, nomeadamente, a qualidade de construção do dispositivo e das ligações, as configurações do protocolo LoRaWAN e as configurações de comunicação física do dispositivo. Um exemplo desta consideração e um cenário possível para novos testes de cobertura seria a utilização de um fator de dispersão de SF12 para a receção de dados no fieldTester, o que iria permitir obter um maior alcance. Estes parâmetros devem ser ajustados de acordo com a necessidade do utilizador e da rede, sendo que a utilização de soluções mais comerciais tipicamente contribui para uma automatização deste processo, através de funcionalidades adaptativas como o ADR.

8. Conclusão e Desenvolvimentos Futuros

Uma análise ao impacto que a IoT tem nas organizações e nas comunidades torna possível afirmar que a evolução deste conceito e das suas tecnologias associadas é de grande importância para a otimização de processos e para uma melhor gestão de recursos. Esta evolução é especialmente importante para a diminuição da complexidade de implementação IoT em serviços públicos, como por exemplo, nos *campus* universitários, onde a IoT demonstra otimizar estes ambientes, tanto a nível académico como a nível estrutural e organizacional. O crescimento da acessibilidade a este tipo de tecnologias é um dos fatores de maior importância na atribuição de uma componente inteligente a infraestruturas e no desenvolvimento constante de conceitos como *smart cities*, *smart homes* ou *smart campus*. Estes conceitos representam conjuntos de diferentes aplicações e tecnologias IoT tipicamente focadas em objetivos gerais semelhantes entre si, como a eficiência na utilização de recursos ou a automatização de processos, mas aplicados a infraestruturas e locais com necessidades específicas. Grande parte destas aplicações baseiam-se em arquiteturas de redes de sensores e tecnologias de comunicação sem fios que permitem a recolha e análise de dados sobre diversos parâmetros físicos e que contribuem para as vantagens mencionadas anteriormente e para tantas outras que a IoT oferece. Estas vantagens foram verificadas nesta dissertação através da implementação direta de uma rede IoT e ainda através da análise de casos de estudo de outros *smart campi*. As redes IoT aplicadas a *smart campus* tipicamente envolvem a utilização de diversas tecnologias de comunicação como meio de interligação entre o *campus* e a Internet e como meio de monitorização e comunicação interna de diversas informações relacionadas com as infraestruturas e com o cotidiano pedagógico. A tecnologia de comunicação LoRa permite que essa comunicação interna seja realizada através de dispositivos de baixo custo e baixo consumo energético capazes de recolher dados sobre diversos parâmetros físicos em qualquer local do *campus*. No contexto de redes de sensores sem fios e de constituição de uma rede deste tipo num *campus* universitário, a tecnologia de comunicação LoRa é a que oferece maior robustez e estabilidade de comunicação associadas a uma alta cobertura e a uma forte componente comunitária de pesquisa e desenvolvimento de soluções LoRa, que contribui para a aprendizagem prática de diversos conceitos de IoT e de comunicação sem fios, como o *Link Budget* ou o RSSI. O estudo destes conceitos e a determinação destes valores contribui para um melhor planeamento das redes e para a análise da propagação dos sinais de comunicação. Essa análise permite avaliar o ambiente de propagação do local de implementação e determinar o impacto da geografia, das condições atmosféricas ou do tipo de obstáculos presentes na qualidade e alcance da comunicação.

O trabalho realizado no âmbito desta dissertação resultou na implementação de uma rede IoT, acedida através de uma *dashboard* na *cloud* e baseada em comunicação LoRa, sendo atualmente constituída por um nó sensor estático que monitoriza o acesso de pessoal docente e um dispositivo de testes que mede a intensidade de sinal recebido em diversos locais de teste. Além da demonstração bem-sucedida da viabilidade de comunicação LoRa ao longo do *campus* e do fluxo bidirecional de dados em toda a rede, os principais resultados obtidos são representados pelos mapas de cobertura e pelo alcance máximo de comunicação de 17,5 km, visto que estes são importantes objetos de análise da tecnologia de comunicação utilizada. A rede IoT implementada e as diferentes ferramentas e tecnologias utilizadas asseguram a escalabilidade da rede e permitem que no futuro sejam facilmente integrados inúmeros dispositivos e servidores, com base no conhecimento descrito nesta dissertação e com base nos manuais de utilização que serão desenvolvidos.

Apesar do sucesso da implementação, foram apontadas algumas possíveis alterações e melhorias da mesma durante o desenvolvimento desta dissertação. Uma das alterações que poderia ter simplificado o processo de desenvolvimento do dispositivo de testes de cobertura seria a utilização do módulo de comunicação LoRa-E5 da Seeed Studio, por este ter a capacidade de comunicar com a placa Arduino através do protocolo UART e por este requer menos ligações físicas. A utilização deste módulo de comunicação iria possibilitar a utilização de uma placa de desenvolvimento ESP32, o que alargaria ainda mais o número de aplicações possíveis, por esta possuir maior poder de processamento e capacidade de comunicação *Bluetooth* e WiFi. A comunicação através do método de ativação OTAA tornaria o processo de recolha de valores de RSSI mais facilitado, sendo que neste método de ativação, as comunicações *uplink* são seguidas de uma comunicação *downlink* de verificação de integridade de dados e o processo de cálculo de RSSI poderia ser feito automaticamente a cada *uplink*. Este processo poderia também ser mais célere e simplificado através da integração de um módulo GPS e de armazenamento no dispositivo de testes. Apesar de não ter sido visivelmente notável, a utilização de fios e de ligações menos robustas resultou num maior ruído de comunicação, o que poderia ser atenuado através do desenvolvimento e construção de uma placa de circuito impresso que aumentaria a eficiência e a eficácia da comunicação LoRa e da comunicação entre componentes.

Entre possíveis desenvolvimentos futuros, englobam-se diversos projetos de aplicação ambiental, como o projeto IPS Sustentável e a criação de espaços verdes mais sustentáveis e monitorizados ou o projeto de monitorização de viveiros de ostras no Estuário do Sado. No âmbito do estudo da tecnologia LoRa, seria também possível a integração de um dispositivo de testes automático num meio de transporte que percorresse diversas zonas de Setúbal e da área circundante, de forma a recolher o maior número de dados e valores de RSSI.

O processo associado de criação de mapas de cobertura poderia também ser automatizado através do desenvolvimento de um *script* Python que recebesse o ficheiro criado pelo dispositivo e apresenta-se os mapas de forma programada e de acordo com os objetivos do estudo. Apesar das vantagens que oferece, a utilização de servidores *cloud* na rede IoT implica uma ligação constante à Internet e a servidores externos à rede do *campus*, o que pode ter implicações na segurança e no funcionamento da rede, uma vez que os servidores não são geridos internamente. De forma a garantir uma maior segurança e o controlo total sobre a gestão da rede, seria possível implementar um servidor de rede LoRa local como o ChirpStack e uma componente aplicacional igualmente local, através de ferramentas como o Node-RED ou aplicações totalmente personalizadas e desenvolvidas no formato mais adequado às necessidades e objetivos de implementação.

Bibliografia

- [1] S. Kumar, P. Tiwari & M. Zymbler, "Internet of Things is a revolutionary approach for future technology enhancement: a review" (2019). Journal of Big data, Volume 6, Issue 1, pp. 1-21. <https://doi.org/10.1186/s40537-019-0268-2>
- [2] S. Kyriazakos, "The Internet of Things" (2015). Resource Management in Future Internet, Chapter 1. <https://books.google.pt/books?id=4P0OBAAAQBAJ>
- [3] P. Gokhale, B. Omkar & B. Sagar, "Introduction to IOT" (2018). International Advanced Research Journal in Science, Engineering and Technology Volume 5, Issue 1, pp. 41-44. <https://doi.org/10.17148/IARJSET.2018.517>
- [4] S. G. H. Soumyalatha, "Study of IoT: understanding IoT architecture, applications, issues and challenges" (2016). 1st International Conference on Innovations in Computing & Net-working (ICICN16), International Journal of Advanced Networking & Applications, Volume 478.
https://www.researchgate.net/publication/330501274_Study_of_IoT_Understanding_IoT_Architecture_Applications_Issues_and_Challenges
- [5] J. Gubbi, B. Rajkumar, M. Slaven & P. Marimuthu, "Internet of Things (IoT): A vision, architectural elements, and future directions" (2013). Future generation computer systems Volume 29, Issue 7, pp. 1645-1660.
<https://doi.org/10.1016/j.future.2013.01.010>
- [6] S. Chen, H. Xu, D. Liu, B. Hu & H. Wang, "A Vision of IoT: Applications, Challenges, and Opportunities With China Perspective" (2014). IEEE Internet of Things Journal, Volume 1, Issue 1, pp. 349-359. <https://doi.org/10.1109/JIOT.2014.2337336>
- [7] J. Wang, M. K. Lim, C. Wang & M. L. Tseng, "The evolution of the Internet of Things (IoT) over the past 20 years" (2021). Computers & Industrial Engineering, Volume 155, pp. 107-174. <https://doi.org/10.1016/j.cie.2021.107174>
- [8] R. Herrero, "Fundamentals of IoT communication technologies" (2022). MOREMEDIA, Part I. <https://link.springer.com/content/pdf/10.1007/978-3-030-70080-5.pdf>
- [9] M. C. Bor, J. Vidler & U. Roedig, "LoRa for the Internet of Things" (2016). European Conference on Wireless Sensor Networks, Volume 16, pp. 361-366.
http://www.researchgate.net/publication/297731094_LoRa_for_the_Internet_of_Things
- [10] S. Al-Sarawi, M. Anbar, K. Alieyan & M. Alzubaidi, "Internet of Things (IoT) communication protocols: Review" (2017). 8th International Conference on Information Technology (ICIT), pp. 685-690. <https://doi.org/10.1109/ICITECH.2017.8079928>
- [11] D. Soni & A. Makwana, "A survey on MQTT: a protocol of Internet of Things (IoT)" (2017). International Conference on Telecommunication, Power Analysis and

Computing Techniques, Volume 20, pp. 173-177.

<https://doi.org/10.1109/TLA.2019.8931137>

- [12] M. Tubaishat & S. Madria “Sensor networks: an overview” (2003). IEEE Potentials, Volume 22, Issue 2, pp. 20-23. <https://doi.org/10.1109/MP.2003.1197877>
- [13] S. Andreev, S. Balandin & Y. Koucheryavy, “Internet of things, smart spaces, and next generation networking” (2012). 13th International Conference NEW2AN, Volume 7469. <https://doi.org/10.1007/978-3-642-40316-3>
- [14] B. Valks, M. H. Arkesteijn, A. Koutamanis & A. C. den Heijer, “Towards a smart campus: supporting campus decisions with Internet of Things applications” (2021). Building Research & Information, Volume 49, Issue 1, pp. 1-20. <https://doi.org/10.1080/09613218.2020.1784702>
- [15] A. Abuarqoub, H. Abusaimh, M. Hammoudeh, D. Uliyan, M. Abu-Hashem, S. Murad, M. Al-Jarrah & F. Al-Fayez, “A Survey on Internet of Things Enabled Smart Campus Applications” (2017). ICFNDS '17: Proceedings of the International Conference on Future Networks and Distributed Systems, Article 50, pp. 1-7. <https://doi.org/10.1145/3102304.3109810>
- [16] S. Al-Sarawi, M. Anbar, R. Abdullah & A. B. Al Hawari, “Internet of Things Market Analysis Forecasts, 2020–2030” (2020). 4th World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4), pp. 449-453. <https://doi.org/10.1109/COMST.2015.2444095>
- [17] F. Xia, L. T. Yang, L. Wang & A. Vinel, “Internet of Things” (2012). International Journal of Communication Systems. <https://doi.org/10.1002/dac.2417>
- [18] R. A. Mouha, “Internet of Things (IoT)” (2021). Journal of Data Analysis and Information Processing, Volume 9, pp. 77-101. <https://doi.org/10.4236/jdaip.2021.92006>
- [19] P. Sethi & S. R. Sarangi, "Internet of Things: Architectures, Protocols, and Applications" (2017). Journal of Electrical and Computer Engineering, Volume 2017, Issue 1. <https://doi.org/10.1155/2017/9324035>
- [20] A.M. Rahmani, S. Bayramov & B. Kiani Kalejahi, “Internet of Things Applications: Opportunities and Threats” (2022). Wireless Personal Communications, Volume 122, pp. 451-476. <https://doi.org/10.1007/s11277-021-08907-0>
- [21] D. Sehwat & N. S. Gill, “Smart Sensors: Analysis of Different Types of IoT Sensors” (2019). 3rd International Conference on Trends in Electronics and Informatics (ICOEI), pp. 523-528. <https://doi.org/10.1109/ICOEI.2019.8862778>
- [22] L. Babun, K. Denney, Z. B. Celik, P. McDaniel & A. S. Uluagac, “A survey on IoT platforms: Communication, security, and privacy perspectives” (2021). Computer Networks, Volume 192, Article n^o 108040. <https://doi.org/10.1016/j.comnet.2021.108040>

- [23] A. Khanna & S. Kaur, "Internet of Things (IoT), Applications and Challenges: A Comprehensive Review" (2020). *Wireless Personal Communications*, Volume 114, pp. 1687–1762. <https://doi.org/10.1007/s11277-020-07446-4>
- [24] F. Zantalis, K. Grigorios, K. Sotiris & K. Dionisis, "A Review of Machine Learning and IoT in Smart Transportation" (2019). *Future Internet*, Volume 11, Issue 4, Article n°94. <https://doi.org/10.3390/fi11040094>
- [25] D. Ushakov, E. Dudukalov, E. Kozlova & K. Shatila, "The Internet of Things impact on smart public transportation" (2022). *Transportation Research Procedia*, Volume 63, pp. 2392-2400. <https://doi.org/10.1016/j.trpro.2022.06.275>
- [26] A. Khanna & R. Anand, "IoT based smart parking system" (2016). 2016 International Conference on Internet of Things and Applications (IOTA), pp. 266-270. <https://doi.org/10.1109/IOTA.2016.7562735>
- [27] G. Lu & Y. J. Yang, "IoT and smart infrastructure" (2017). *Internet of Things and Data Analytics Handbook*, pp. 481-493. <https://doi.org/10.1002/9781119173601.ch29>
- [28] C. Paul, A. Ganesh & C. Sunitha, "An overview of IoT based smart homes" (2018). 2nd International Conference on Inventive Systems and Control (ICISC), pp. 43-46. <https://doi.org/10.1109/ICISC.2018.8398858>
- [29] B. Farahani, F. Firouzi & K. Chakrabarty, "Healthcare iot" (2020). *Intelligent Internet of Things: From Device to Fog and Cloud*, pp. 515-545. https://doi.org/10.1007/978-3-030-30367-9_11
- [30] N. Scarpato, P. Alessandra, L. Di Nunzio & F. Fallucchi, "E-health-IoT universe: A review" (2017). *International Journal on Advanced Science, Engineering and Information Technology*, Volume 7. <https://doi.org/10.18517/ijaseit.7.6.4467>
- [31] N. Isakadze & Seth S. Martin, "How useful is the smartwatch ECG?" (2020). *Trends in Cardiovascular Medicine*, Volume 30, Issue 7, pp. 442-448. <https://doi.org/10.1016/j.tcm.2019.10.010>
- [32] W. Saltzstein, "Bluetooth Wireless Technology Cybersecurity and Diabetes Technology Devices" (2020). *Journal of Diabetes Science and Technology*, Volume 14, Issue 6. <https://doi.org/10.1177/1932296819864416>
- [33] N. Vijayakumar & R. Ramya, "The real time monitoring of water quality in IoT environment" (2015). 2015 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), pp. 1-5. <https://doi.org/10.1109/ICIIECS.2015.7193080>
- [34] E. Broday & M. Gameiro da Silva, "The role of internet of things (IoT) in the assessment and communication of indoor environmental quality (IEQ) in buildings: a review" (2022). *Smart and Sustainable Built Environment*, Volume 12, Issue 3, pp. 584-606. <https://doi.org/10.1108/SASBE-10-2021-0185>
- [35] M. N. Hassan, M. R. Islam, F. Faisal, F. H. Semantha, A. H. Siddique & M. Hasan, "An IoT based Environment Monitoring System" (2020). 3rd International Conference

- on Intelligent Sustainable Systems, pp. 1119-1124.
<https://doi.org/10.1109/ICISS49785.2020.9316050>
- [36] M. S. Mahoud & A. H. Mohamad, "A study of efficient power consumption wireless communication techniques/modules for internet of things (IoT) applications" (2016). Advances in Internet of Things, Volume 6, pp. 19-29.
<http://dx.doi.org/10.4236/ait.2016.62002>
- [37] R. Sánchez-Corcuera, A. Nuñez-Marcos, J. Sesma-Solance, "Smart cities survey: Technologies, application domains and challenges for the cities of the future" (2019). International Journal of Distributed Sensor Networks, Volume 15, Issue 6.
<https://doi.org/10.1177/1550147719853984>
- [38] I. de Almeida, L. Mendes, J. Rodrigues & M. da Cruz, "5G Waveforms for IoT Applications" (2019). IEEE Communications Surveys & Tutorials, Volume 21, Issue 3, pp. 2554-2567. <https://doi.org/10.1109/COMST.2019.2910817>
- [39] Y. Niu, Y. Li & D. Jin, "A survey of millimeter wave communications (mmWave) for 5G: opportunities and challenges" (2015). Wireless Networks, Volume 21, pp. 2657-2676. <https://doi.org/10.1007/s11276-015-0942-z>
- [40] R. Ford, M. Zhang, M. Mezzavilla, S. Dutta, S. Rangan & M. Zorzi, "Achieving Ultra-Low Latency in 5G Millimeter Wave Cellular Networks" (2017). IEEE Communications Magazine, Volume 55, Issue 3, pp. 196-203.
<https://doi.org/10.1109/MCOM.2017.1600407CM>
- [41] J. Cao, M. Ma, H. Li, R. Ma, Y. Sun, P. Yu & L. Xiong, "A Survey on Security Aspects for 3GPP 5G Networks" (2020). IEEE Communications Surveys & Tutorials, Volume 22, Issue 1, pp. 170-195. <https://doi.org/10.1109/COMST.2019.2951818>
- [42] M. Shafi, A. F. Molisch, P. J. Smith, T. Haustein, P. Zhu, P. De Silva, F. Tufvesson, A. Benjebbour, & G. Wunder, "5G: A Tutorial Overview of Standards, Trials, Challenges, Deployment, and Practice" (2017). IEEE Journal on Selected Areas in Communications, Volume 35, Issue 6, pp. 1201-1221.
<https://doi.org/10.1109/JSAC.2017.2692307>
- [43] J. A. Adebisola, A. A. Ariyo, O. A. Elisha, A. M. Olubunmi & O. O. Julius, "An Overview of 5G Technology" (2020). 2020 International Conference in Mathematics, Computer Engineering and Computer Science (ICMCECS), pp. 1-4.
<https://doi.org/10.1109/ICMCECS47690.2020.240853>
- [44] L. Li, H. Xiaoguang, C. Ke & H. Ketai, "The applications of WiFi-based Wireless Sensor Network in Internet of Things and Smart Grid" (2011). 6th IEEE Conference on Industrial Electronics and Applications, pp. 789-793.
<https://doi.org/10.1109/ICIEA.2011.5975693>
- [45] E. Mozaffariahrar, F. Theoleyre, M. Menth, "A Survey of Wi-Fi 6: Technologies, Advances, and Challenges" (2022). Future Internet, Volume 14, pp. 293.
<https://doi.org/10.3390/fi14100293>

- [46] Y. Daldoul, D. E. Meddour & A. Ksentini, "Performance evaluation of OFDMA and MU-MIMO in 802.11ax networks" (2020). *Computer Networks*, Volume 182, Article n° 107477. <https://doi.org/10.1016/j.comnet.2020.107477>
- [47] M. T. Suer, P. Jose, & H. Tchouankem, "Experimental Evaluation of IEEE 802.11ax - Low Latency and High Reliability with Wi-Fi 6?" (2022). *GLOBECOM 2022 - IEEE Global Communications Conference*, pp. 377-382. <https://doi.org/10.1109/GLOBECOM48099.2022.10001475>
- [48] C. BasuMallick, "What Is WiFi 6? Meaning, Speed, Features, and Benefits" (2023). *Networking, Spiceworks*. <https://www.spiceworks.com/tech/networking/articles/what-is-wifi-six/>
- [49] J. Sun & X. Zhang, "Study of ZigBee Wireless Mesh Networks" (2009). *9th International Conference on Hybrid Intelligent Systems*, Volume 2, pp. 264-267. <https://doi.org/10.1109/HIS.2009.164>
- [50] M. S. Pan & Y. C. Tseng, "ZigBee and their applications" (2007). *Sensor networks and configuration: Fundamentals, standards, platforms, and applications*, pp. 349-368. https://link.springer.com/content/pdf/10.1007/3-540-37366-7_16.pdf
- [51] R. Robles-Enciso, I. P. Morales-Aragón, A. Serna-Sabater, M. T. Martínez-Inglés, A. Mateo-Aroca, J. M. Molina-Garcia-Pardo & L. Juan-Llácer, "LoRa, Zigbee and 5G Propagation and Transmission Performance in an Indoor Environment at 868 MHz" (2023). *Sensors*, Volume 23, Issue 6, Article n° 3283. <https://www.mdpi.com/1424-8220/23/6/3283>
- [52] C. M. Ramya, M. Shanmugaraj & R. Prabakran, "Study on ZigBee technology" (2011). *3rd International Conference on Electronics Computer Technology*, Volume 6, pp. 297-301. <https://doi.org/10.1109/ICECTECH.2011.5942102>
- [53] K. E. Jeon, J. She, P. Soonsawad & P. C. Ng, "BLE Beacons for Internet of Things Applications: Survey, Challenges, and Opportunities" (2018). *IEEE Internet of Things Journal*, Volume 5, Issue 2, pp. 811-828. <https://doi.org/10.1109/JIOT.2017.2788449>
- [54] J. F. Dian, A. Yousefi & S. Lim, "A practical study on Bluetooth Low Energy (BLE) throughput" (2018). *IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, pp. 768-771. <https://doi.org/10.1109/IEMCON.2018.8614763>
- [55] K. Cho, W. Park, M. Hong, G. Park, W. Cho, J. Seo & K. Han, "Analysis of Latency Performance of Bluetooth Low Energy (BLE) Networks" (2015). *Sensors*, Volume 15, Issue 1, pp. 59-78. <https://www.mdpi.com/1424-8220/15/1/59>
- [56] R. Ratasuk, B. Vejlgard, N. Mangalvedhe, A. Ghosh, "NB-IoT system for M2M Communications" (2016). *IEEE Wireless Communications and Networking Conference*, pp. 1-5. <https://doi.org/10.1109/WCNC.2016.7564708>
- [57] B. Martinez, F. Adelantado, A. Bartoli & X. Vilajosana, "Exploring the Performance Boundaries of NB-IoT" (2019). *IEEE Internet of Things Journal*, Volume 6, Issue 3, pp. 5702-5712. <https://doi.org/10.1109/JIOT.2019.2904799>

- [58] R. S. Sinha, Y. Wei & S. H. Hwang, "A survey on LPWA technology: LoRa and NB-IoT" (2017). *ICT Express*, Volume 3, Issue 1, pp. 14-21.
<https://doi.org/10.1016/j.ict.2017.03.004>
- [59] A. Augustin, J. Yi, T. Clausen & W. M. Townsley, "A Study of LoRa: Long Range & Low Power Networks for the Internet of Things" (2016). *Sensors*, Volume 16, Issue 9, Article n° 1466. <https://www.mdpi.com/1424-8220/16/9/1466>
- [60] F. Zulkifli, "The Power of Chirp Spread Spectrum (CSS): History, Techniques, and Applications" (2024). *HamRadio, LoRa*. <https://hamradio.my/2024/09/the-power-of-chirp-spread-spectrum-css-history-techniques-and-applications/>
- [61] I. K. Enriko, F. Gustiyana, K. Kurnianingsih & E. Sari, "LoRaWAN for Smart Street Lighting Solution in Pangandaran Regency" (2023). *JOIV : International Journal on Informatics Visualization*, Issue 7, pp. 2452. <https://doi.org/10.30630/joiv.7.4.01198>
- [62] J. Petajarvi, K. Mikhaylov, M. Pettissalo, J. Janhunen & J. Linatti, "Performance of a low-power wide-area network based on LoRa technology: Doppler robustness, scalability, and coverage" (2017). *International Journal of Distributed Sensor Networks*, Volume 13, pp. 1-16. <https://doi.org/10.1177/1550147717699412>
- [63] A. Pötsch & F. Hammer, "Towards End-to-End Latency of LoRaWAN: Experimental Analysis and IIoT Applicability" (2019). *IEEE International Workshop on Factory Communication Systems (WFCS)*, pp. 1-4.
<https://doi.org/10.1109/WFCS.2019.8758033>
- [64] A. Almarhabi, A. J. Aljohani & M. Moinuddin, "LoRa and High-Altitude Platforms: Path Loss, Link Budget and Optimum Altitude" (2020). *8th International Conference on Intelligent and Advanced Systems (ICIAS)*, pp. 1-6.
<https://doi.org/10.1109/ICIAS49414.2021.9642705>
- [65] S. Sundaram, J. Prasanna, W. Du & Z. Zhao, "A Survey on LoRa Networking: Research Problems, Current Solutions, and Open Issues" (2020). *IEEE Communications Surveys & Tutorials*, Volume 22, Issue 1, pp. 371-388.
<https://doi.org/10.14569/IJACSA.2019.0101106>
- [66] D. Rani & N. S. Gill, "Review of various IoT standards and communication protocols" (2019). *International Journal of Engineering Research and Technology*, Volume 12, Issue 5, pp. 647-657.
https://www.researchgate.net/publication/369771563_Review_of_Various_IoT_Standards_and_Communication_Protocols
- [67] S. Hamdani & H. Sbeyti, "A Comparative study of COAP and MQTT communication protocols" (2019). *7th International Symposium on Digital Forensics and Security (ISDFS)*, pp. 1-5. <https://doi.org/10.1109/ISDFS.2019.8757486>
- [68] U. Hunkeler, H. L. Truong & A. Stanford-Clark, "MQTT-S - A publish/subscribe protocol for Wireless Sensor Networks" (2008). *3rd International Conference on Communication Systems Software and Middleware and Workshops*, pp. 791-798.
<https://doi.org/10.1109/COMSWA.2008.4554519>

- [69] B. Mishra & A. Kertesz, "The Use of MQTT in M2M and IoT Systems: A Survey" (2020). IEEE Access, Volume 8, pp. 201071-201086.
<https://doi.org/10.1109/ACCESS.2020.3035849>
- [70] J. Yick, B. Mukherjee & D. Ghosal, "Wireless sensor network survey" (2008). Computer Networks, Volume 52, Issue 12, pp. 2292-2330.
<https://doi.org/10.1016/j.comnet.2008.04.002>
- [71] M. Kocakulak & I. Butun, "An overview of Wireless Sensor Networks towards internet of things" (2017). IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC), pp. 1-6. <https://doi.org/10.1109/CCWC.2017.7868374>
- [72] Y. Chung, "An energy-efficient unicast routing protocol for wireless sensor networks" (2011). International Journal of Computer Science & Emerging Technologies, Volume 2, Issue 1, pp. 60-64.
https://www.researchgate.net/publication/366001339_IJCSET_V21
- [73] H. Gong, L. Fu, X. Fu, L. Zhao, K. Wang & X. Wang, "Distributed multicast tree construction in wireless sensor networks" (2016). IEEE Transactions on Information Theory, Volume 63, Issue 1, pp. 280-296. <https://doi.org/10.1109/TIT.2016.2623317>
- [74] H. Jeong, J. Kim & Y. Yoo, "Adaptive Broadcasting Method Using Neighbor Type Information in Wireless Sensor Networks" (2011). Sensors, Volume 11, pp. 5952-5967.
<https://doi.org/10.3390/s110605952>
- [75] P. Tyagi & S. Jain, "Comparative study of routing protocols in wireless sensor network" (2012). International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 9.
<https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=f6caa6e5ecd2c3f9088799e2b1497d93dc6754ed>
- [76] LoRa Alliance Official Website. <https://lora-alliance.org/>
- [77] J. Haxhibeqiri, E. De Poorter, I. Moerman & J. Hoebeke, "A Survey of LoRaWAN for IoT: From Technology to Application" (2018). Sensors, Volume 18, Issue 11, Article n° 3995. <https://www.mdpi.com/1424-8220/18/11/3995>
- [78] Y. Song, J. Lin, M. Tang & S. Dong, "An Internet of Energy Things Based on Wireless LPWAN" (2017). Engineering, Volume 3, Issue 4, pp. 460-466.
<https://doi.org/10.1016/J.ENG.2017.04.011>
- [79] "Global LoRa and LoRaWAN IoT Market Size, Share & Industry Trends Analysis Report By End User, By Component (Hardware, Platform and Services), By Application, By Regional Outlook and Forecast, 2023 - 2030" (2022). KBV research, ID: KBV-15976. <https://www.kbvresearch.com/lora-and-lorawan-iot-market/>
- [80] M. Saari, A. M. bin Baharudin, P. Sillberg, S. Hyrynsalmi & W. Yan, "LoRa – A survey of recent research trends" (2018). 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), pp. 872-877. <https://ieeexplore.ieee.org/abstract/document/8400161>

- [81] M. O Farooq & D. Pesch, "Analyzing LoRa: A use case perspective" (2018). IEEE 4th World Forum on Internet of Things (WF-IoT), pp. 355-360.
<https://doi.org/10.1109/WF-IoT.2018.8355224>
- [82] S. Devalal & A. Karthikeyan, "LoRa Technology - An Overview" (2018). 2nd International Conference on Electronics, Communication and Aerospace Technology (ICECA), pp. 284-290. <https://doi.org/10.1109/ICECA.2018.8474715>
- [83] "LoRa Technology, It's Features and Application" (2017). Techplayon, IoT.
<https://www.techplayon.com/lora-technology-benefits-application/>
- [84] F. Kidwai, A. Madaan, S. Bansal & A. Sahu, "Peer-to-Peer Communication Using LoRa Technology" (2021). International Conference on Innovative Computing and Communications. Advances in Intelligent Systems and Computing, Volume 1165.
https://doi.org/10.1007/978-981-15-5113-0_52
- [85] P. S. Cheong, J. Bergs, C. Hawinkel & J. Famaey, "Comparison of LoRaWAN classes and their power consumption" (2017). IEEE Symposium on Communications and Vehicular Technology (SCVT), pp. 1-6.
<https://doi.org/10.1109/SCVT.2017.8240313>
- [86] M. Centenaro, C. Costa, F. Granelli, C. Sacchi & L. Vangelista, "A Survey on Technologies, Standards and Open Challenges in Satellite IoT" (2021). IEEE Communications Surveys & Tutorials, pp. 1.
<https://doi.org/10.1109/COMST.2021.3078433>
- [87] N. Min-Allah & S. Alrashed, "Smart campus - A sketch" (2020). Sustainable Cities and Society, Volume 59, Article n^o 102231. <https://doi.org/10.1016/j.scs.2020.102231>
- [88] A. Abuarqoub, H. Abusaimh, M. Hammoudeh, M. Uliyan, M. Abu-Hashem, S. Murad, M. Al-Jarrah & F. Alfayez, "A Survey on Internet of Things Enabled Smart Campus Applications" (2017). International Conference on Future Networks & Distributed Systems, pp. 1-7. <https://doi.org/10.1145/3102304.3109810>
- [89] S. Pervez, S. ur Rehman & G. Alandjani, "Role of internet of things (iot) in higher education" (2018). Proceedings of ADVED, pp. 792-800.
https://www.researchgate.net/publication/328420304_ROLE_OF_INTERNET_OF_THINGS_IOT_IN_HIGHER_EDUCATION
- [90] E. Gilman, S. Tamminen, R. Yasmin, E. Ristimella, E. Peltonen, M. Harju, L. Lovén, J. Riekkki, S. Pirttikangas, "Internet of Things for Smart Spaces: A University Campus Case Study" (2020). Sensors, Volume 20, Issue 13, Article n^o 3716.
<https://www.mdpi.com/1424-8220/20/13/3716>
- [91] T. Domínguez-Bolaño, V. Barral, C. J. Escudero & J. A. García-Naya, "An IoT system for a smart campus: Challenges and solutions illustrated over several real-world use cases" (2024). Internet of Things, Volume 25, Article n^o 101099.
<https://doi.org/10.1016/j.iot.2024.101099>
- [92] Módulo de comunicação LoRa SX1262, Semtech Products.
<https://www.semtech.com/products/wireless-rf/lora-connect/sx1262>

- [93] Módulo de comunicação LoRa SX1276, Semtech Products.
<https://www.semtech.com/products/wireless-rf/lora-connect/sx1276>
- [94] Datasheet do módulo de comunicação LoRa RFM95, HopeRF Eletronic.
https://cdn.sparkfun.com/assets/learn_tutorials/8/0/4/RFM95_96_97_98W.pdf
- [95] Módulo de comunicação LoRa SX1301, Semtech Products.
<https://www.semtech.com/products/wireless-rf/lora-core/sx1301>
- [96] Nó sensor ERS2 CO2, Elsys.se. <https://www.elsys.se/shop/product/ers-co2-2-0/>
- [97] Nó sensor Aranet4 PRO, Aranet. <https://aranet.com/en/pro/products/aranet4-pro>
- [98] Nó sensor LDS02 Door Sensor, Dragino.
<https://www.dragino.com/products/lorawan-nb-iot-door-sensor-water-leak/item/181-lds02.html>
- [99] Gateway Conduit IP67 Base Station, Multitech. <https://multitech.com/all-products/cellular/cellular-gateways/conduit-ip67-base-station/>
- [100] Gateway Aranet PRO Base Station, Aranet.
<https://aranet.com/en/pro/products/aranet-pro-12-base-station>
- [101] Gateway Wirnet iStation, Kerlink. <https://www.kerlink.com/product/wirnet-istation/>
- [102] Getting started, The Things Stack.
<https://www.thethingsindustries.com/docs/getting-started/the-things-stack-basics/>
- [103] The Things Stack Sandbox Dashboard. <https://eu1.cloud.thethings.network>
- [104] ChirpStack Official Website. <https://www.chirpstack.io/>
- [105] Thingpark Enterprise, Actility. <https://www.actility.com/enterprise-iot-connectivity-solutions/>
- [106] ThingWorx, PLC. <https://www.ptc.com/en/products/thingworx>
- [107] Home Assistant Official Website. <https://www.home-assistant.io/>
- [108] TagoIO Official Website. <https://tago.io/>
- [109] Mapa topográfico de Portugal. Topographic-map. <https://pt-pt.topographic-map.com/map-6nmzs/Portugal/>
- [110] Mapa 3D. Google Earth. <https://earth.google.com/web/>
- [111] J. José, "RESENS - REDE DE SENSORES PARA O BOSQUE DO IPS" (2024). Projeto Final de Curso de Licenciatura em Engenharia Eletrotécnica e de Computadores, ESTS/IPS.
https://drive.google.com/file/d/1nq_htN5WNYazUIrtqJp74w1tQRB2k562
- [112] R. André, "LoRaGrid - Sistema de monitorização de consumos elétricos LoRa" (2024). Dissertação de Mestrado em Engenharia Eletrotécnica e de Computadores, ESTS/IPS.
https://drive.google.com/file/d/1M1X7pA47d8_nm3A08o3aWcGARk8aOGy
- [113] Datasheet do módulo Bluetooth HC-06.
<https://www.olimex.com/Products/Components/RF/BLUETOOTH-SERIAL-HC-06/resources/hc06.pdf>

- [114] Datasheet do sensor de temperatura DS18B20.
<https://www.analog.com/media/en/technical-documentation/data-sheets/ds18b20.pdf>
- [115] TTN Mapper Official Website. <https://ttnmapper.org>
- [116] P. Pinto, "Recorde: Comunicação com Lora chega aos 1336 km (Portugal)" (2023). High Tech, pplware. <https://pplware.sapo.pt/high-tech/recorde-comunicacao-com-lora-chega-aos-1336-km-portugal/>

Anexo

Este anexo é constituído por duas partes onde se apresenta na íntegra o código utilizado para programar o dispositivo de testes de cobertura (Anexo 1) e para desenvolver a aplicação móvel utilizada para recolher dados de geolocalização e associá-los a valores de RSSI (Anexo 2).

Anexo 1: Código do fieldTester – Dispositivo para testes de cobertura equipado com RFM95

```
/*
 * Code for a LoRa development field test device designed for testing LoRa
communication and applications
 * Used in the "Campus of Things" project linked to "Escola Superior de
Tecnologia - Polytechnic Institute of Setúbal"
 * Field test device composed by the following equipment: Arduino Uno, RFM95
(LoRa), HC-06 (Bluetooth) and DS18B20 (Temperature)
 * Copyright (c) 2024 - Bernardo Milheiro
*/
// - Bluetooth
#include <SoftwareSerial.h>
// Sets the RX and TX pins connected to HC-06.
SoftwareSerial bluetooth(8, 9); // (RX, TX)

// - Temperature
#include <OneWire.h>
#include <DallasTemperature.h>
// DS18B20 Temperature Sensor Setup.
OneWire oneWire(4);
DallasTemperature temperatureObj(&oneWire);

// - LoRa
#include <lmic.h>
#include <hal/hal.h>
#include <SPI.h>
// Network Session Key, App Session Key and Device Address Setup.
static const PROGMEM u1_t NWKSKEY[16] = { 0xFE, 0x09, 0xA6, 0xE3, 0xA4, 0x89,
0x87, 0x66, 0x8C, 0xE8, 0x06, 0xCF, 0xAA, 0xCF, 0x39, 0x36 };
static const u1_t PROGMEM APPSKEY[16] = { 0x77, 0x63, 0xF5, 0xF5, 0x9A, 0x6D,
0x9D, 0x23, 0x8C, 0xDE, 0xAE, 0x97, 0x58, 0x57, 0x74, 0x1A };
static const u4_t DEVADDR = 0x260BD6F8;
// Declares job variable.
static osjob_t sendjob;
// Sets message max size and initial message as "Hello".
uint8_t dataSend[5];
String dataString = "Hello";
// Sets interval of transmission in seconds.
const unsigned TX_INTERVAL = 20;
// Declares rssi variable.
int16_t rssi = 0;
```

```

// Pin mapping RFM95.
const lmic_pinmap lmic_pins = {
  .nss = 10,
  .rxtx = LMIC_UNUSED_PIN,
  .rst = 5,
  .dio = { 2, 3, LMIC_UNUSED_PIN },
};

// RSSI Calculation Function.
int16_t getRssi() {
// Returns correct RSSI (dBm) value of the last received packet.
// Calculation per SX1276 datasheet rev.7 §5.5.5.
#define RSSI_OFFSET 64
#define SX1276_FREQ_LF_MAX 525000000
#define SX1272_RSSI_ADJUST -139
#define SX1276_RSSI_ADJUST_LF -164
#define SX1276_RSSI_ADJUST_HF -157
  int16_t snr = (LMIC.snr * 10) / 40;
  int16_t rssiTemp = LMIC.rssi - RSSI_OFFSET;
  int16_t rssiAdjust;
  if (LMIC.freq > SX1276_FREQ_LF_MAX) {
    rssiAdjust = SX1276_RSSI_ADJUST_HF;
  } else {
    rssiAdjust = SX1276_RSSI_ADJUST_LF;
  }
  int16_t packetRssi = LMIC.rssi + 125 - RSSI_OFFSET;
  if (snr < 0) {
    rssiTemp = rssiAdjust + packetRssi + snr;
  } else {
    rssiTemp = rssiAdjust + (16 * packetRssi) / 15;
  }
  return rssiTemp;
}

// Temperature Reading Function.
void getTemp() {
  // Get temperature value from sensor and assigns that data.
  temperatureObj.requestTemperatures();
  float temperature = temperatureObj.getTempCByIndex(0);
  // If temperature is within valid measurements.
  if (temperature > -55 && temperature < 125) {
    // Turns float temperature value into String.
    dataString = String(temperature);
  } else {
    // Prints and assigns "Failure" as data.
    Serial.println(F("Temperature (DS18B20) reading failure."));
    dataString = "Failure";
  }
}

```

```

}

// Data Transmission Function.
void do_send(osjob_t* j) {
    digitalWrite(13, LOW); // Turns the LED off
    // Check if there's any current TX/RX job running.
    if (LMIC.opmode & OP_TXRXPEND) {
        Serial.println(F("There is a current TX/RX job running"));
    } else {
        // Upstream data transmission at the next possible time.
        dataString.toCharArray(dataSend, dataString.length() + 1);
        LMIC_setTxData2(1, dataSend, dataString.length(), 0);
    }
}

// Event Behaviour Function.
void onEvent(ev_t ev) {
    // Gets time tag.
    os_getTime();
    // Switch function for occurred event. For development purposes, only
    "EV_TXCOMPLETE" event was considered for the system behaviour.
    switch (ev) {
        // If transmission is successful.
        case EV_TXCOMPLETE:
            // Prints sent data.
            Serial.println(F("\n-----"));
            Serial.println(F("Device name: fieldTester"));
            Serial.print(F("Data Sent: "));
            Serial.println(dataString);
            Serial.println(F("EV_TXCOMPLETE (includes waiting for RX windows)"));
            // Prints received data if its size is different than 0.
            if (LMIC.dataLen) {
                digitalWrite(13, HIGH); // Turns the LED on
                rssi = getRssi(); // Gets last downlink RSSI value.
                Serial.print(F("Data Received: "));
                Serial.write(LMIC.frame + LMIC.dataBeg, LMIC.dataLen);
                Serial.print(F("\n"));
            }
            // Prints RSSI value if it's between 0 and -150 (valid values) and sends it
            via Bluetooth.
            if (rssi < 0 && rssi > -150) {
                Serial.print(F("Last downlink RSSI: "));
                Serial.print(rssi);
                bluetooth.print(rssi);
            }
            // Gets current temperature value.
            getTemp();
            // Schedule next transmission.

```

```

        os_setTimedCallback(&sendjob, os_getTime() + sec2osticks(TX_INTERVAL),
do_send);
        break;
        // In case other event occurs prints a message.
        default:
            Serial.println(F("Expected event didn't ocorred!"));
            break;
    }
}

////////////////////////////////////
// - Setup & Loop    //
////////////////////////////////////
void setup() {
    // Serial, Bluetooth and Temperature Sensor initialization.
    Serial.begin(9600);
    bluetooth.begin(9600);
    temperatureObj.begin();
    // Initialize digital pin 13 as an output.
    pinMode(13, OUTPUT);
    // LMIC initialization.
    os_init();
    // Reset the MAC state. Session and pending data transfers will be discarded.
    LMIC_reset();
    // Set static session parameters.
    uint8_t appskey[sizeof(APPSKEY)];
    uint8_t nwkskey[sizeof(NWKSKEY)];
    memcpy_P(appskey, APPSKEY, sizeof(APPSKEY));
    memcpy_P(nwkskey, NWKSKEY, sizeof(NWKSKEY));
    LMIC_setSession(0x1, DEVADDR, nwkskey, appskey);
    // Set up the channels used by the Things Network.
    LMIC_setupChannel(0, 868100000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI);
    LMIC_setupChannel(1, 868300000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI);
    LMIC_setupChannel(2, 868500000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI);
    LMIC_setupChannel(3, 867100000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI);
    LMIC_setupChannel(4, 867300000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI);
    LMIC_setupChannel(5, 867500000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI);
    LMIC_setupChannel(6, 867700000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI);
    LMIC_setupChannel(7, 867900000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI);
    LMIC_setupChannel(8, 868800000, DR_RANGE_MAP(DR_FSK, DR_FSK), BAND_MILLI);
    // Disable link check validation.
    LMIC_setLinkCheckMode(0);
    // Gateway is configured to use SF9 and SF12 for its RX2 window.
    LMIC.dn2Dr = DR_SF9;
    // Set data rate and transmit power for uplink.
    LMIC_setDrTxpow(DR_SF9, 7);
    // Start job.
    do_send(&sendjob);
}

```

```
}  
  
void loop() {  
  os_runloop_once();  
}
```

Anexo 2: Código de blocos da aplicação móvel - ApplInventor

```
initialize global long to
initialize global lat to
initialize global rssi to
create empty list

when LocationSensor1 .LocationChanged
do
  latitude longitude altitude speed
  set global lat to get latitude
  set global long to get longitude
  set latitudeValue .Text to get global lat
  set longitudeValue .Text to get global long

when Screen1 .Initialize
do
  if call MFile1 .Exists
    path file:///storage/emulated/0/Documents/data.csv
  then
    call Notifier1 .ShowAlert
    notice Data will be added to existing file.
  else
    call MFile1 .SaveFile
    text
    fileName file:///storage/emulated/0/Documents/data.csv
    call Notifier1 .ShowAlert
    notice Data file created.

when devicePicker .BeforePicking
do
  set devicePicker .Elements to BluetoothClient1 .AddressesAndNames

when Clock1 .Timer
do
  if BluetoothClient1 .IsConnected
  then
    if call BluetoothClient1 .BytesAvailableToReceive > 0
    then
      set global rssi to call BluetoothClient1 .ReceiveText
      numberOfBytes call BluetoothClient1 .BytesAvailableToReceive
      set rssiValue .Text to get global rssi
      call Notifier1 .ShowAlert
      notice Data recieved.
    else
      set rssiValue .Text to No Bluetooth data.
    end
  else
    set Status .TextColor to
    set Status .Text to Status: Disconnected
    call Notifier1 .ShowAlert
    notice Bluetooth Connection failed!
    set Clock1 .TimerEnabled to false

when devicePicker .AfterPicking
do
  set devicePicker .Selection to call BluetoothClient1 .Connect
  address devicePicker .Selection
  if BluetoothClient1 .IsConnected
  then
    set Clock1 .TimerEnabled to true
    set Status .Text to Status: Connected
    set Status .TextColor to
    call Notifier1 .ShowAlert
    notice Bluetooth Connected!
  else
    call Notifier1 .ShowAlert
    notice Bluetooth Connection failed!
```

```
when getData .Click
do
  if
    length get global rssi ≥ 3 and contains text get global rssi piece "-"
  then
    call MFile1 .AppendToFile
      text join ["", get global lat, ",", get global long, ",", get global rssi, "."]
      fileName "file:///storage/emulated/0/Documents/data.csv"
    call Notifier1 .ShowAlert
      notice "GPS and RSSI saved!"
  else
    set rssiValue :Text to "Invalid Bluetooth data."
```