



## INTERNET DAS COISAS E EFICIÊNCIA ENERGÉTICA EM AGRICULTURA DE PRECISÃO

PUNGO, ERICA. (1); TEIXEIRA, EMANUEL (2); REIS, ANTONIO D. (3); VELEZ,  
FERNANDO J. (4)

(1) Instituto de Telecomunicações e UBI, E-mail: [erica.pungo@ubi.pt](mailto:erica.pungo@ubi.pt)

(2) Instituto de Telecomunicações e UBI, E-mail: [emanuelt@ubi.pt](mailto:emanuelt@ubi.pt)

(3) Universidade da Beira Interior, E-mail: [adreis@ubi.pt](mailto:adreis@ubi.pt)

(4) Instituto de Telecomunicações e UBI, UBI, E-mail: [fjv@ubi.pt](mailto:fjv@ubi.pt)

### RESUMO

Este trabalho apresenta aspetos de Redes de Sensores sem Fios aplicadas à promoção da agricultura de precisão, inteligente e autónoma. O sistema de irrigação aqui desenvolvido com base em conceitos da Internet das Coisas (IoT) destina-se a combater os desperdícios, que ocorrem num modelo de rega tradicional e aprimorar a eficiência energética nas comunicações. Para desenvolver este sistema de irrigação foi utilizado o Raspberry Pi 4, juntamente com os módulos XBEE – PRO S1 da norma do Institute of Electrical and Electronic Engineering (IEEE) 802.15.4. Testaram-se os alcances de comunicação pretendidos. As medições da humidade do terreno, condições atmosféricas e horários apropriados de rega fornecem dados que são processados pelo sistema de monitorização, com o objetivo de realizar uma rega inteligente, autónoma, eficaz e sem desperdícios. Considerou-se a plataforma de interface gráfica Blynk para criar um painel que permite a visualização dos níveis de humidade e temperatura. A configuração do sistema de rega envolve a utilização de uma mini bomba submersível em água, de um sensor de temperatura e de um sensor de humidade do solo ligados a um Arduino Uno. Este sensor comunica os valores da humidade para o Raspberry Pi. Os resultados subjacentes comprovam as vantagens da solução e mostram que a eficiência da comunicação entre nós XBEE é diferenciada dependendo da distância. Através da comparação de dois modos de operação (Application Programming Interface, API, e modo transparente, AT) conclui-se que o modo AT é mais fácil de utilizar e económico, pois realiza configurações diretas e comunicações básicas.

**Palavras-chave:** Rega, Raspberry Pi, XBEE-PRO, Agricultura de precisão.

## ABSTRACT

*This work presents aspects of Wireless Sensor Networks applied to the promotion of precision, intelligent and autonomous agriculture. The irrigation system developed in this work based on Internet of Things (IoT) concepts aims at avoiding waste that occur in traditional watering, whilst increasing energy efficiency in communications. To develop this irrigation system, the Raspberry Pi 4 was used, together with the XBEE – PRO S1 modules of the Institute of Electrical and Electronic Engineering (IEEE) 802.15.4 standard, to test the desired communication ranges. Measurements of soil humidity, atmospheric conditions and appropriate irrigation schedules provide data that is processed by the monitoring system, to carry out intelligent, autonomous, effective, and waste-free irrigation. One considered the Blynk graphical interface platform to create a dashboard that allows the visualization of the humidity and temperature levels. The configuration of the irrigation system involves the use of a mini pump submersible in water, temperature sensor, and a soil humidity sensor connected to an Arduino Uno. These humidity sensor reports provide the humidity and temperature values to the Raspberry Pi. Underlying results prove the advantages of the proposed solution and show that the efficiency of the communication between XBEE nodes is different depending on the distance. It was possible to compare the two modes of operation (Application Programming Interface, API, and Transparent Mode, AT) whilst realizing that the AT mode is easier and economical to use, as it enables direct configurations and basic communications.*

**Keywords:** Watering, Raspberry Pi 4, XBEE-PRO S1, Precision Agriculture.

## 1 INTRODUÇÃO

Os dispositivos de recolha de dados IoT energeticamente eficientes podem ser utilizados no campo agrícola para recolher dados em tempo real através de ligações sem fios. Estes dispositivos IoT foram otimizados para vários cenários agrícolas e podem ser utilizados para analisar dados de culturas, solo, gado e clima [1], no contexto da agricultura de precisão.

O sistema de irrigação moderno tem evoluído significativamente com a integração de tecnologias de comunicação sem fio. Nesse contexto, o uso dos módulos XBEE-PRO S1 em conjunto com o Raspberry Pi oferece uma solução para a automação e controle remoto de sistemas de irrigação. Através dessa comunicação sem fio eficiente, o Raspberry Pi pode transmitir comandos precisos para acionar e regular o sistema de irrigação, permitindo uma gestão mais precisa e conveniente dos recursos hídricos. Essa integração entre dispositivos promove não apenas a eficiência no uso da água, mas também a flexibilidade e adaptabilidade do sistema, proporcionando uma abordagem sustentável e tecnologicamente avançada para a agricultura e o manuseio de áreas verdes.

Os módulos XBEE-PRO foram projetados para atender a norma IEEE 802.15.4 e suportar as necessidades exclusivas de redes de sensores sem fio de baixo custo e baixo consumo de energia. Estes dispositivos requerem energia mínima e fornecem entrega confiável de dados entre dispositivos. Operam dentro da banda de frequência ISM 2,4 GHz e são compatíveis pino a pino entre si [2].

O Raspberry Pi é um microcomputador, que contém, num hardware de baixo custo, todas as funções que estamos habituados a ver num computador. O modelo Raspberry Pi 4, tem WiFi e Bluetooth integrados, que disponibilizam boa conectividade com o que o leitor necessitar, inclui um processador Quad Core de 1,5 GHz e entre 2 e 6 GB de memória RAM [3].

## **2 DESENVOLVIMENTO**

### **2.1 Eficiência Energética das RSSFs**

A eficiência energética é um dos aspetos importantes a considerar no desenvolvimento de nós sensores e da rede. Por exemplo, as escolhas feitas na camada física de um nó sensor afeta o consumo de energia de todo o dispositivo e a conceção de protocolos de nível mais alto [4].

A eficiência energética das RSSFs corresponde à habilidade de se realizar uma ou várias tarefas com uma quantidade menor de recursos energéticos [5].

As técnicas de economia de energia concentram-se em dois subsistemas, o subsistema de rede (a gestão de energia é direcionada para as operações em cada nó único) e o subsistema de deteção (onde técnicas são utilizadas para reduzir a quantidade de amostras com elevado consumo de energia) [5].

### **2.2 Procedimentos Experimentais**

A realização deste trabalho de desenvolvimento, envolveu os seguintes procedimentos.

#### **Configuração do Raspberry Pi**

Seguindo os passos sugeridos pelo autor de [4], começou-se por montar e configurar o Raspberry Pi, inicialmente utilizou-se um teclado, um rato e um ecrã, conexão Ethernet, cartão micro SD, deu-se início a instalação do Raspberry Pi OS que foi um processo simples e rápido onde o leitor responde um conjunto de questões básicas; por predefinição, as credenciais de acesso do Raspberry Pi OS são: Utilizador: pi; Password: raspberry.

Para identificar o endereço IP que nos permite trabalhar remotamente no Raspberry Pi, utiliza-se o comando "sudo ip addr show". Existem duas formas principais de fixar o endereço IP:

- 1-** Utilizar DHCP (este geralmente é dinâmico).

**2-** Fixar o IP no router, conhecido por IP estático; usado o comando “sudo nano /etc/network/interfaces” é aberto um ficheiro em edição que precisa-se substituir de acordo o IP desejado.

Se a opção for utilizar o DHCP utiliza-se “iface etho inet dhcp”. Se o IP for o estático utiliza-se “iface etho inet static”, “address IP”, “netmask”, “gateway”; também se pode utilizar a alternativa de acesso via WiFi, utilizando “iface wlan0 inet static”, wpa-ssid “nome da rede”, wpa-psk “password”.

Após a configuração da rede podemos usar remotamente o Raspberry Pi usando o PUTTY e VNC viewer, através da conexão por SSH. Para iniciar sessão ao PUTTY deve-se introduzir o endereço IP, número do Porto, e seleccionar o tipo de conexão. E no VNC precisa-se apenas do endereço de IP.

Alguns comandos usados para instalar o servidor VNC no Raspberry Pi: “sudo apt-get install tightvncserver” e depois executar: “/usr/bin/tightvncserver”.

## Desenvolvimento Do Sistema de Irrigação

Como se observa na Figura 1, a montagem do sistema de irrigação, iniciou-se conectando os sensores de humidade e temperatura ao Arduino:

- **Conexão:** o pino VCC do sensor de humidade foi ligado a 5 V no Arduino, o GND ao GND e o A0 a uma porta analógica (como A0); definiu-se o pino 7 do Arduino como um pino de 5V, que foi ligado ao VCC do sensor de temperatura, o respetivo GND ao GND, e o pino de saída analógica foi ligado ao A1 do Arduino.

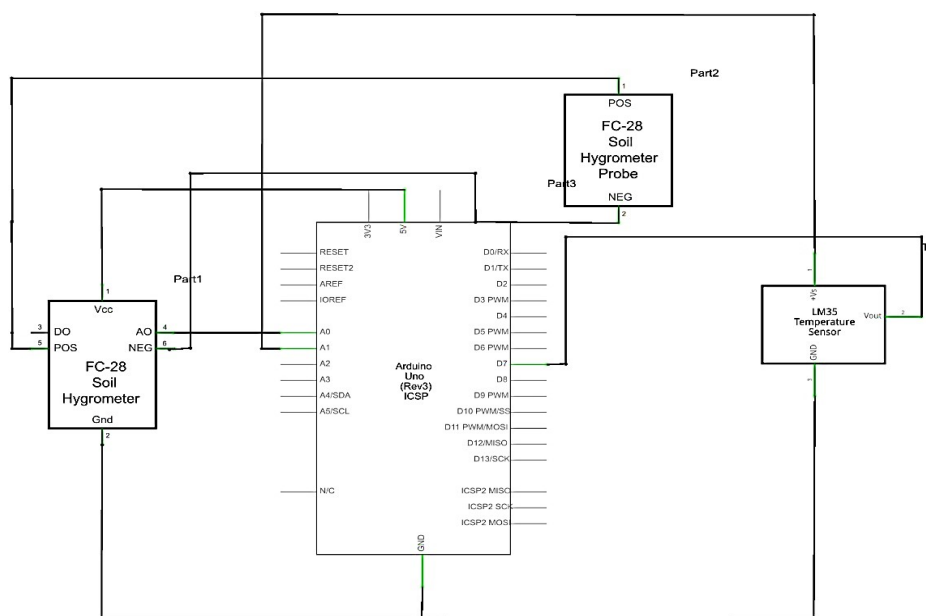


Figura 1-Esquema elétrico da montagem no Arduino.

- **Definição dos limites de programação:** Com base nas leituras, estabeleceram-se limites para determinar quando será conveniente regar ou não. Se o valor da leitura da humidade for <50% e a temperatura estiver entre 4 a 25 °C, o sistema informa que o solo necessita de água. Todas estas informações são enviadas através da comunicação *serial* para o Raspberry Pi.
- **Observações:** Neste contexto o Arduíno serviu apenas como um conversor intermediário entre os sensores de humidade/temperatura e o Raspberry Pi. Assim, o Arduíno apenas lida com a leitura analógica do sensor, enquanto o Raspberry Pi é dedicado ao processamento, ao armazenamento e na comunicação com serviços externos, ou mesmo à exibição dos dados.

A montagem com o Raspberry Pi envolve os seguintes passos:

- **Conexão:** O pino VCC do relé é ligado ao pino de 5 V do Raspberry PI, o GND ao GND, e IN ao pino 12 (PWM), como se apresenta na Figura 2. O relé foi ligado em modo normalmente aberto. Atrui-se-lhe a responsabilidade de acionar a minibomba de água submersível. Como a sua alimentação nominal é de 6 V foi necessário incluir uma fonte de alimentação compatível para habilitar a sua capacidade básica de funcionamento.
- **Configuração da Comunicação Serial:** configura-se a leitura dos dados do Arduíno através da porta serial USB, da seguinte forma: `comunicacaoSerial = serial.Serial('/dev/ttyACM0', 9600)`. Configura-se uma comunicação *serial* com o dispositivo conectado à porta através da utilização de `/dev/ttyACM0` a uma taxa de transmissão de 9600 b/s.
- **Controle do Relé:** o relé, inicialmente em nível baixo, é colocado em nível alto se o valor recebido for de humidade < 50% e a temperatura estiver no intervalo entre 4 a 25 °C; mas se não for esta condição, o relé mantém no estado baixo.

## Comunicação Com Os Modulos Xbee-Pro S1

As comunicações sem fios proporcionam flexibilidade e mobilidade, permitindo assim a troca de dados entre dispositivos em áreas remotas ou de difícil acesso. A interconexão entre o Raspberry Pi e os módulos XBEE-PRO S1 é realizada por meio das portas séries disponíveis no dispositivo, como se apresenta em [5].

Através da biblioteca Python "serial", é possível estabelecer uma comunicação bidirecional confiável, permitindo enviar e receber dados entre os nós da rede. A configuração para comunicação dos módulos XBEE-PRO S1 começou-se com a ligação dos módulos ao computador utilizando um adaptador USB da Libellium. O software XCTU deteta automaticamente os módulos XBEE-PRO S1 conectados como se observa na Figura 3 [5].

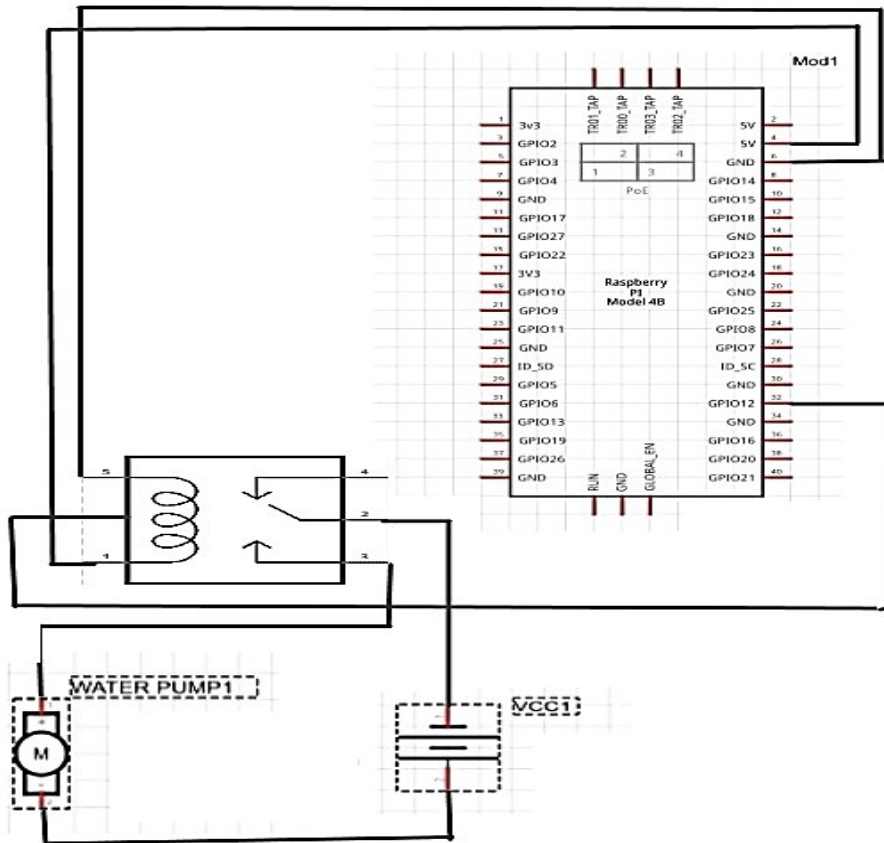


Figura 2-Esquema de ligação no Raspberry Pi

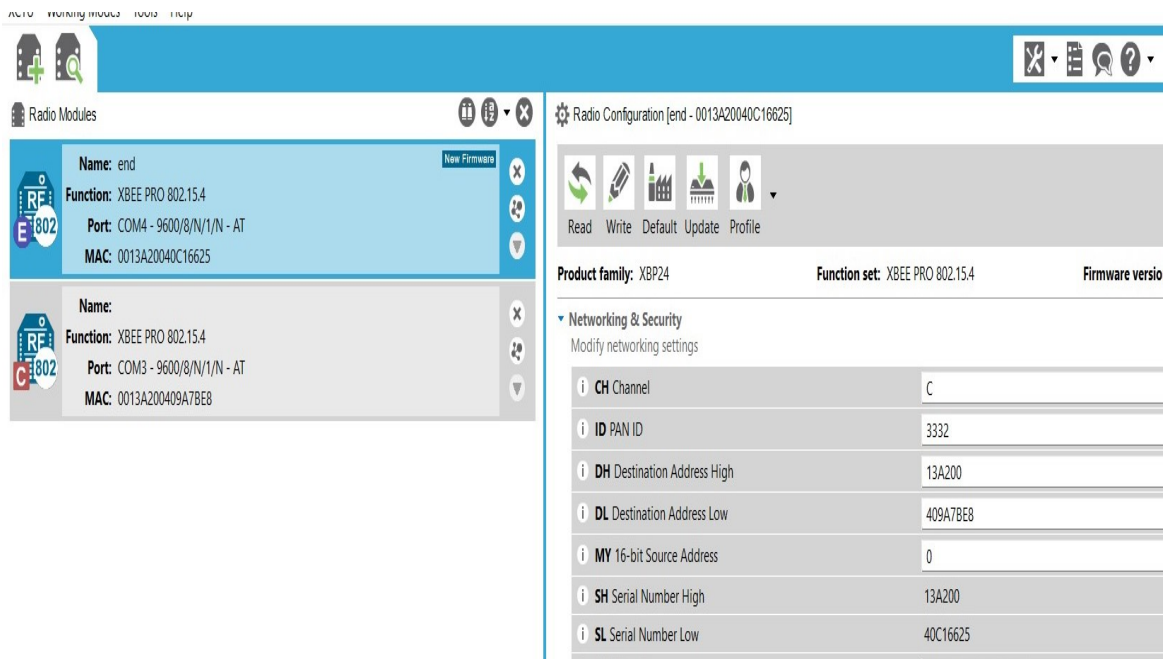
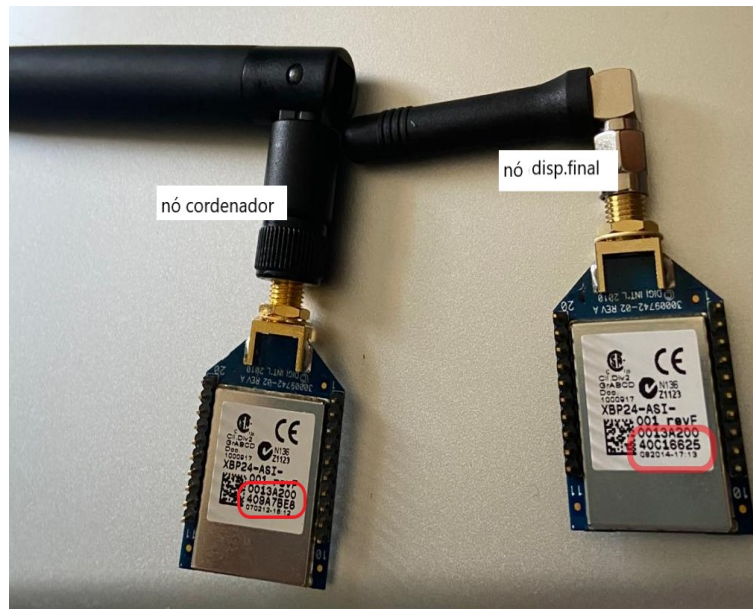


Figura 3-Configuração dos módulos no XCTU (foi selecionada a família XBP24; conjunto de funções XBEE-PRO 802.15.4, e a versão de firmware 10ee).

OS passos comuns para configurar os módulos XBEE-PRO S1 no modo AT são os seguintes. Deve-se garantir que os módulos tenham o mesmo PAN ID: 3332, e haja correspondência dos *destination address High* (DH)

e *destination address Low* (DL) dos endereços da camada MAC, como se apresenta na Figura 4. O DH e o DL devem trocar-se entre nós, ou seja, o endereço MAC do coordenador deve ser a concatenação do DH (os primeiros 8 dígitos) e do DL (os segundos 8 dígitos) do nó de dispositivo final. Por outro lado, o endereço MAC do nó de dispositivo final deverá ser a concatenação dos DH e DL do nó coordenador.



**Figura 4-Endereços dos nós coordenador e dispositivo final.**

A taxa de transmissão da interface *serial* para comunicação entre a porta *serial* do módulo e o *host* designa-se por BD. O parâmetro RO define o atraso entre caracteres necessários antes do início da transmissão. NB configura a paridade para o UART e o AP serve para habilitar o API.

Depois da configuração, a rede estará pronta a realizar a comunicação entre os sensores, em que um é o nó coordenador e outro é o nó fonte (dispositivo final).

Para que o Xbee coordenador envie os valores obtidos pelo script em Python para módulo Xbee *end-device*, utilizou-se a comunicação *serial* conectada ao módulo Xbee, envolvendo os seguintes passos:

Contemplam-se ainda os seguintes passos:

**Configurar a Comunicação Serial:** Configura-se a comunicação *serial* para a porta onde o módulo XBEE está conectado. Isso envolve definir a taxa de transmissão (baud rate) e outros parâmetros.

```
"ser = serial.Serial('/dev/ttyUSB0', baudrate=9600)"
```

**Enviar valores:** Depois de ter a comunicação *serial* configurada, utiliza-se o comando `"ser.write(message.encode())"` para enviar dados para o módulo XBEE-PRO S1.

Executou-se o código no para certificar que o Xbee coordenador esteja a enviar os dados lidos ao End-device

A Figura 5 apresenta o protótipo do sistema de irrigação completo, compreendendo também comunicação dos módulos XBEE-PRO S1.



Figura 5-Prototipo do sistema de irrigação completo.

### Painel De Visualização

Foi criado um painel de visualização na plataforma *Blynk*, para controlar dentro e fora do local que se opera.

A plataforma *Blynk* por ser conhecida por conectar dispositivos via Internet e na nuvem, utiliza alguns métodos de segurança tais como [ 3 ] :

- **Autenticação:** Exige que o utilizador autentique o seu dispositivo e aplicações utilizando um Token de autenticação para que só os dispositivos autorizados se conectem a plataforma.
- **Comunicação encriptada:** Para evitar interações não autorizadas.
- **Controlo de acessos:** Capacidade de controlar quem pode aceder aos seus projetos.
- **Proteção de dados:** Políticas de privacidade para protecção de dados dos utilizadores.

A Figura 6 apresenta o fluxograma do sistema, após a leitura dos sensores de humidade e de temperatura, comunica ao Raspberry Pi e envia os dados no painel de interface *Blynk*, o Raspberry Pi compara os valores segundo aos critérios estabelecidos e mediante aos critérios liga ou desliga a bomba, e o nó coordenador comunica ao dispositivo final, espera durante um tempo para retomar o ciclo novamente. Os detalhes podem ser consultados em [6].

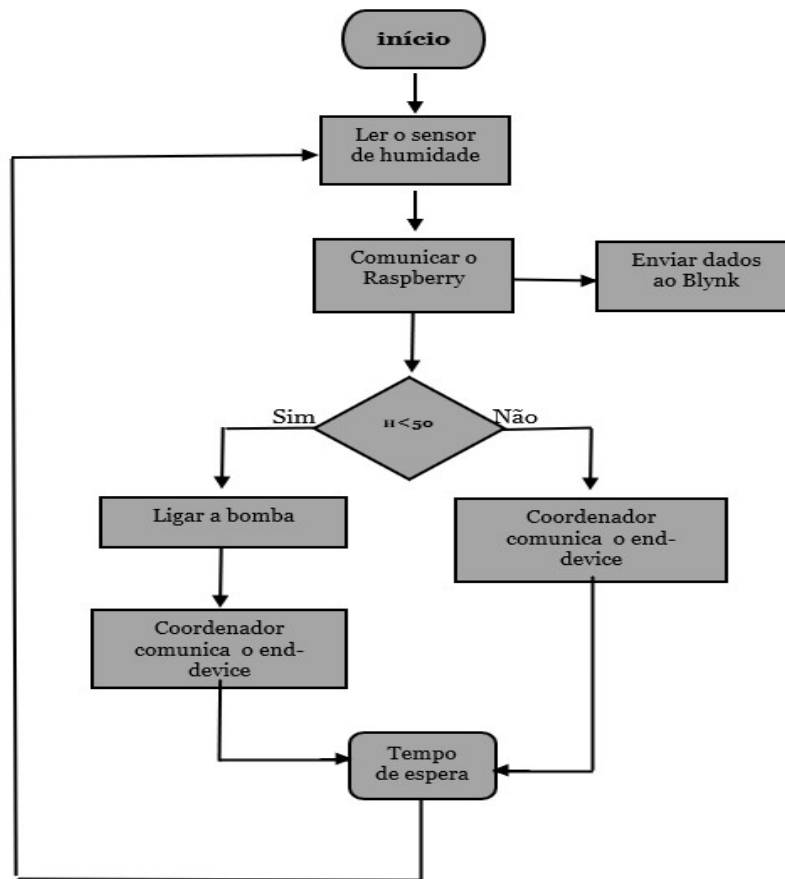


Figura 6-Fluxograma do sistema.

### 3 RESULTADOS

#### 3.1 Consumo de Energia

A Figura 7 apresenta sistema composto por dois nós:

- um é o nó coordenador, que lê as informações vindas dos sensores, por meio do Raspberry Pi, passando também dados ao painel;
- e o outro é o dispositivo final, que recebe as informações vinda do coordenador.

A Tabela1 apresenta os dispositivos utilizados e suas respectivas taxas de consumo de energia (P), bem como o número de horas (T) que cada dispositivo está ligado ou em utilização. Com base nesses números, é possível calcular quantidade total de energia (E) consumida por cada dispositivo ao longo de uma hora.

$$E = P * T \quad (1)$$

O consumo total estimado de energia foi de 2,84 Wh (ou seja, 10368 J). Este valor representa uma quantidade relativamente reduzida de energia em comparação com muitos outros dispositivos eletrônicos. É importante considerar estes valores ao projetar sistemas eletrônicos, especialmente quando se analisa a eficiência energética.

Além da estimativa do consumo de energia no período de uma hora, utilizou-se o Wattímetro para saber o consumo real de energia do sistema, como se apresenta na Figura 8.

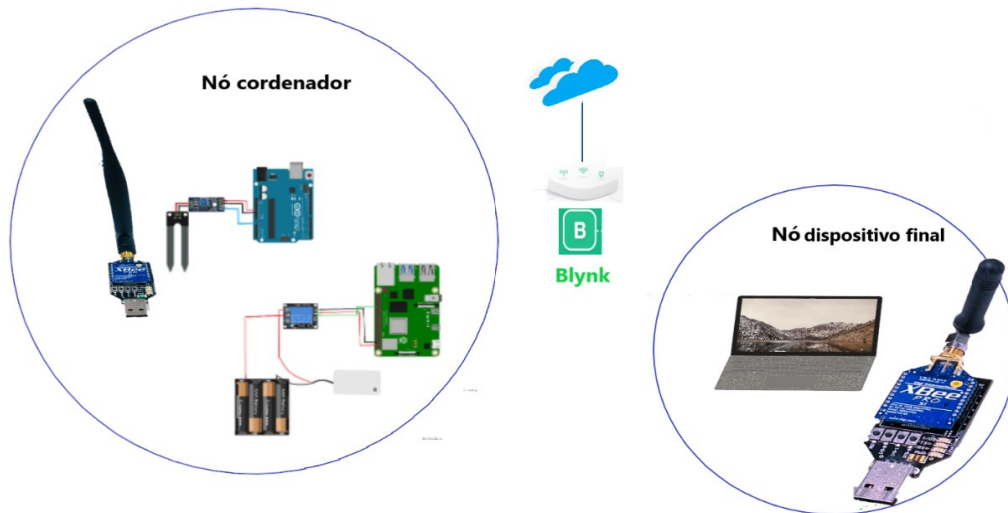


Figura 7-Illustração do sistema completo de comunicação.

Tabela 1 – Consumo de Energia [6].

Dispositivo	Potência (W)	Tempo(h)	Energia (Wh)
Arduíno	0,03	1	0,03
Raspberry Pi	2,7	1	2,7
Relé	0,005	1	0,005
Sensor H	0,005	1	0,005
XBEE-PRO S1	0,1	1	0,1
Sensor T	$0,09 \cdot 10^{-3}$	1	$0,09 \cdot 10^{-3}$



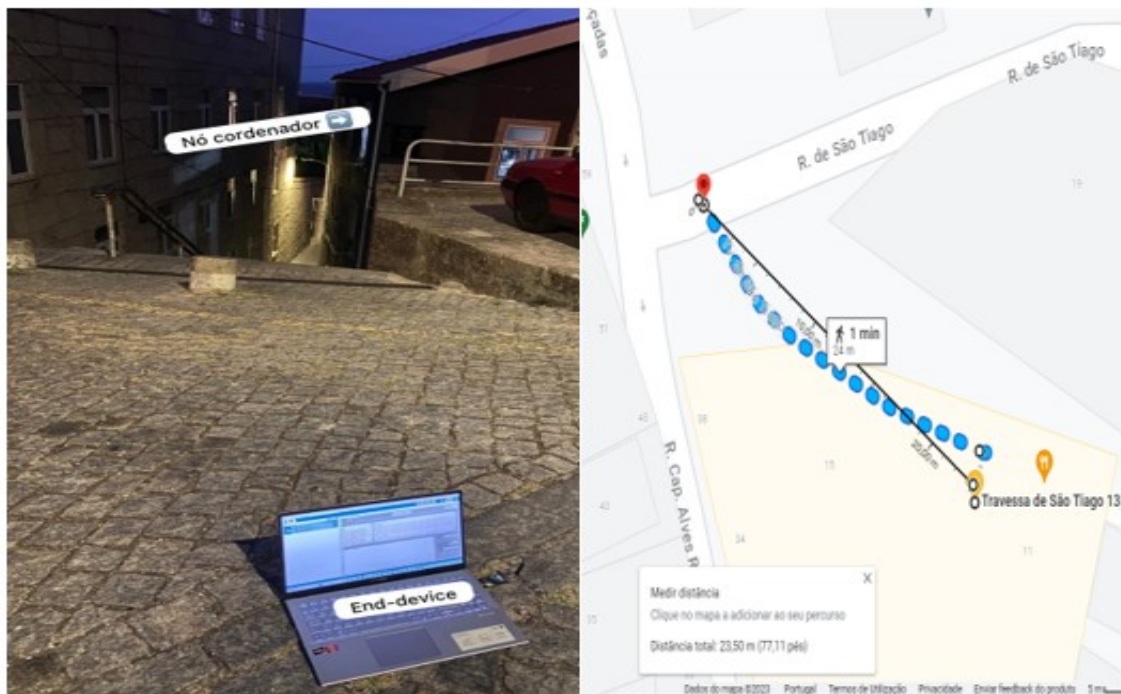
Figura 8-Consumo de energia vista pelo Wattímetro.

Com uma duração total de 15 horas e 46 minutos, o sistema consumiu uma energia total de 54,11 Wh, correspondendo uma potência de 3,5 W, que é um valor de consumo aceitável e que se aproxima do valor esperado.

Existe no entanto uma ligeira diferença entre o cálculo estimado e o real, devido ao sensor de temperatura, visto que esse é um dispositivo que fornece uma saída analógica proporcional à temperatura. Este facto tem como consequência que, mediante o período de tempo que a temperatura varia, a tensão também varia. Por este motivo, o valor do consumo real pode ser ligeiramente diferente do consumo estimado.

### 3.2 Testes de Alcance

Foram efetuados dois testes de alcance em locais diferentes. No primeiro teste, no ambiente da Figura 9, o nó coordenador estava dentro de um compartimento e a distância entre o nó de dispositivo final e o coordenador era separado por uma parede; os resultados indicam que em zonas urbanas a comunicação é mais eficaz a curtas distâncias, com uma resposta confiável e eficiente obtida entre 10 e 20 metros. No entanto, conforme a distância aumenta, a confiabilidade e a velocidade da comunicação diminuem consideravelmente. Isso é evidenciado pelos testes bem-sucedidos entre 20 e 30 metros, mas com respostas consideravelmente demoradas, o que não é mal segundo a norma IEEE 802.15.4.



**Figura 9-Primeiro teste de alcance dos nós na Traversa de São Tiago 13, na Covilhã.**

O segundo teste, foi realizado na UBIMedical, onde colocamos o nó coordenador e o dispositivo final em espaço livre sem barreira nenhuma, como se apresenta na Figura 10.

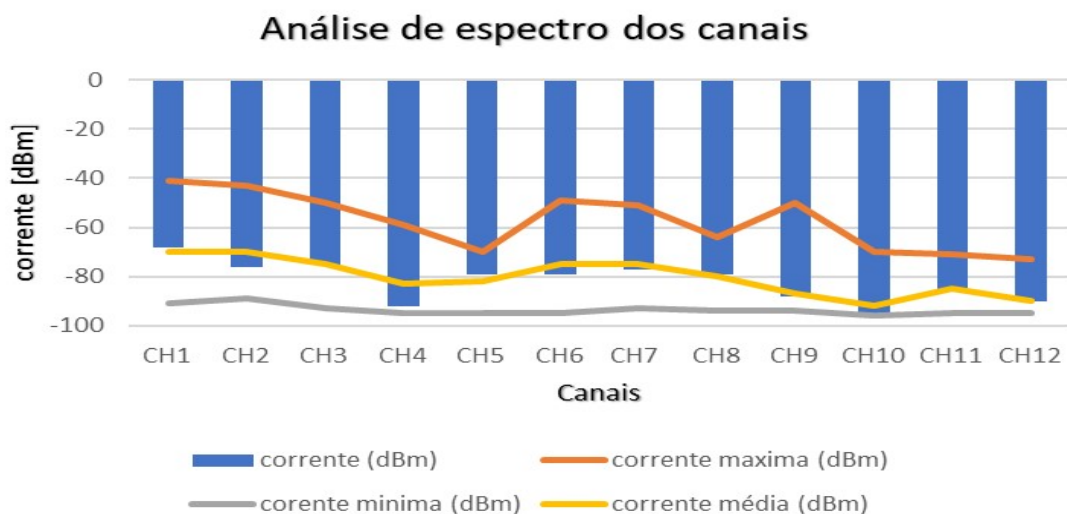


**Figura 10-Segundo teste de alcance dos nós, na UBIMedical.**

Os resultados indicam que quando os nós são distribuídos sem barreiras nenhuma, a comunicação é mais eficaz para distâncias até cerca de 60 metros, sem perda de mensagens. Já para distâncias superiores a 60 metros, constatou-se a perda de dados e atraso na recepção dos dados. A Figura 10 também apresenta a distância de teste com eficácia.

### 3.3 Análise dos Canais

Como se apresenta nas Figuras 11, o software XCTU, permitiu observar o espectro das bandas de rádio de 12 canais, possibilitando a observação do nível da intensidade do sinal dos canais, incluindo medições de máxima, mínima e média para ajudar na melhor configuração da rede. Quanto maiores forem os valores das correntes em dBm, melhor o sinal do canal.



**Figura 11-Análise do espectro dos canais na banda ISM (a 2,4 GHz).**

Os resultados apresentados na Figura 11 mostram que o CH1 demonstrou ser o melhor canal, pelo valor máximo de intensidade atingido, o que o torna numa conexão mais forte e estável.~

### Medição do RSSI e Taxa de Transferência de Dados

O Received Signal Strength Indication (RSSI) indica a intensidade do sinal recebido, é expressa em dBm, e é utilizado para avaliar a qualidade do sinal de rádio ou a potência do sinal em uma conexão de comunicação sem fio. O RSSI é dado pela seguinte formula:

$$RSSI = 10 \log \frac{Pr}{Pref} \quad (3)$$

onde  $Pr$  é a potência recebida, e  $P_{ref}$  é a potencia recebida por referência e, normalmente, tem o valor de 1 mW. Para se saber o valor RSSI, o XCTU dá-nos a opção *Range Test*. O valor medido de RSSI é obtido após a transferência de 100 pacotes de 34 bytes cada e feita a média do RSSI de cada transmissão. Após a configuração dos dados geram-se gráficos, como se apresentam nas Figuras 12 e 13.

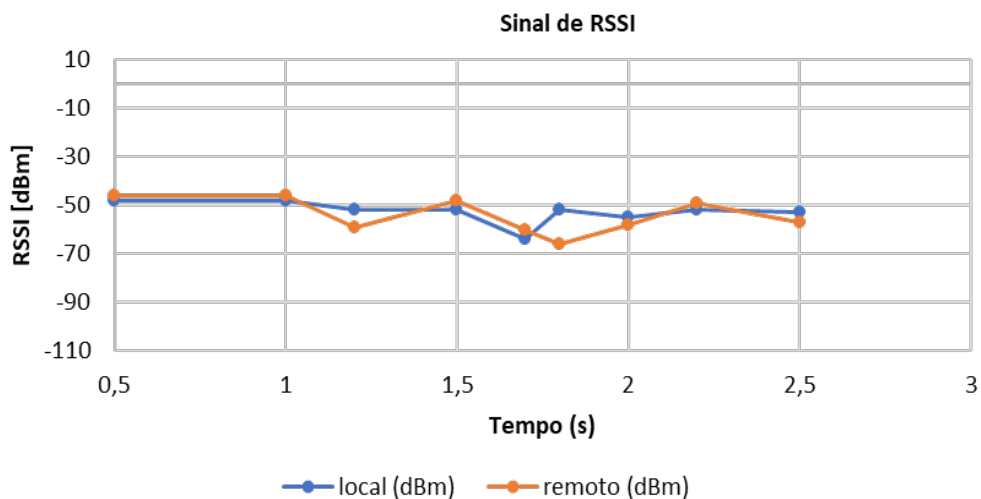


Figura 12-Monitorização dos valores medidos do RSSI.

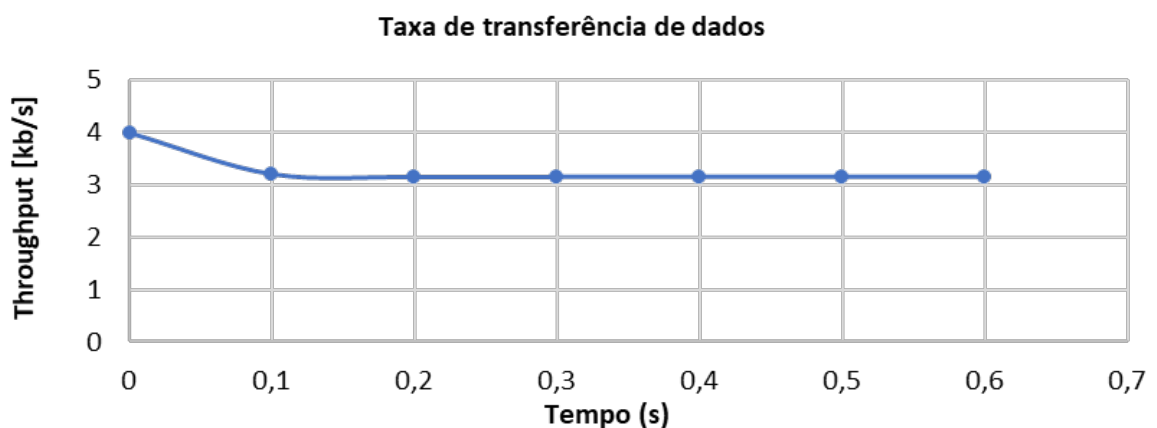


Figura 13-Evolução da taxa de transferência de dados ao longo do tempo.

Os valores de RSSI obtidos dependem do local, e da distância. Neste caso, os valores de RSSI do nó local e do nó remoto obtidos estão compreendidos em valores espetáveis que se aproximam ao valor médio, e constata-se a perda de pacotes. Em alguns momentos, o valor de RSSI chega a -50 dBm, o que corresponde a uma qualidade do sinal aceitável, ou seja, “boa” (e quanto maior melhor).

A taxa de transferência de dados é o número de bits transferidos por unidade de tempo. Para saber o valor da taxa de transferência, o XCTU dá-nos a opção *throughput*; foi utilizado um intervalo de um minuto de transmissão e pacotes de 100 bytes. Como podemos observar na Figura 13, a taxa de transferência média foi de 3,15 kb/s, este teste foi feito a pouco mais de 1,5 metros de distância, em local fechado.

### Painel de Visualização

Como resultado do painel criado na plataforma *Blynk*, esta interface permite fazer a leitura dos dados da humidade e temperatura em tempo real tornando-os disponíveis para acompanhamento e análise. A interface também traz a possibilidade de ser instalada em smartphones, como se apresenta na Figura 14.

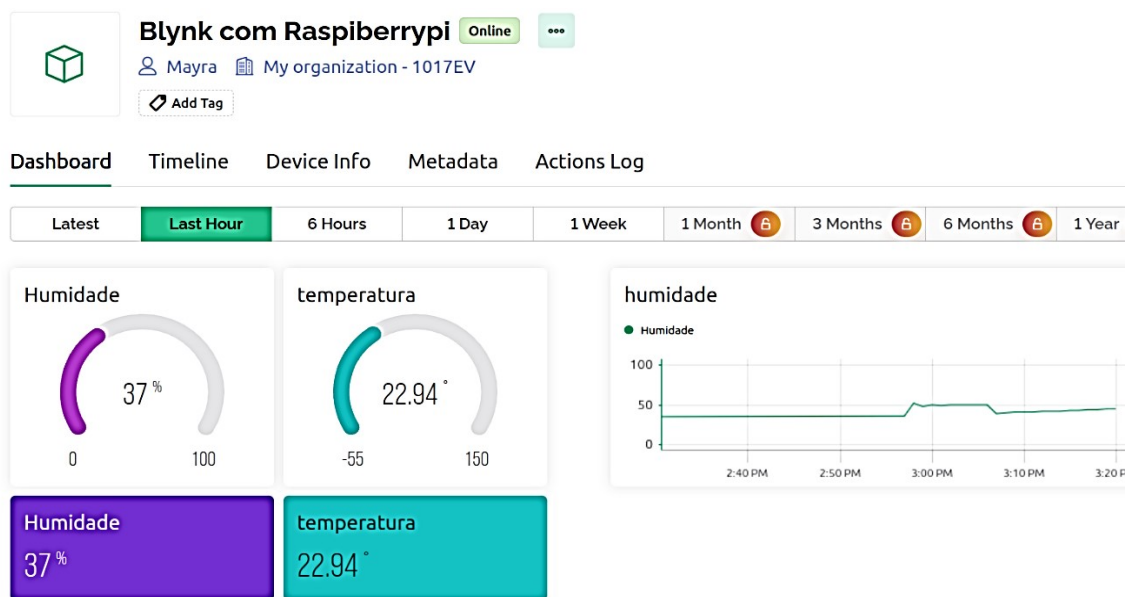


Figura 14- Apresentação do painel no Blynk.

Os testes efetuados na camada de aplicação da plataforma de controlo e gestão de dispositivos IoT, foram realizados com sucesso, desde o registo até à autenticação, conectividade e segurança na comunicação via HTTP.

## 4 CONCLUSÕES

Enquanto que o valor real do consumo de energia no período de uma hora foi de 3,5 Wh, o seu valor estimado foi de 2,84 Wh. A diferença resulta principalmente da diferença entre os valores estimado e medido

pelo sensor de temperatura, pois existe dependência da tensão relativamente à temperatura real medida.

Quanto aos resultados da comunicação sem fios, o valor medido de RSSI mais elevado foi de -50 dBm, que corresponde a uma “boa” qualidade de sinal. A taxa de transferência de dados média foi de 3,15 kb/s, um valor aceitável, dado que foram enviados dados simples, como leituras de humidade e de temperatura.

Como o *Blynk* é muito simples de utilizar, o desenvolvimento da interface criada foi eficiente. A solução tem a vantagem de permitir monitorizar os parâmetros através de um *smartphone*, de qualquer forma, em qualquer momento e lugar, desde que haja ligação à Internet. A plataforma *Blynk*, utilizada para conectar dispositivos via internet e na nuvem, utiliza alguns métodos de segurança, como o *Token* de autenticação para que só os dispositivos autorizados se conectem à plataforma, com comunicação encriptada, controlo de acessos e proteção de dados.

O sistema de rega desenvolvido controla a humidade do solo, evitando assim danos e desperdícios de água e energia. Pode ser utilizado não só em agricultura, mas em casas, hotéis, e até se estudado e aprofundado em mais detalhe, servirá de base para prevenção e combate de incêndios, que são muito comuns na zona centro de Portugal. Sugere-se que no futuro se possa adoptar a tendência mMTC da 5G, utilizando equipamentos de comunicação de baixa energia, pequenos volumes de dados e custo reduzido, e também que o fornecimento de energia possa vir a ser realizado através de um sistema híbrido solar e eólico.

A tecnologia 5G permite a ligação entre máquinas (ou dispositivos) em tempo real e ainda a sua gestão, de maneira eficiente, promovendo tomadas de decisões precisas, redução de custos de operação, segurança de dados e eficiência em todo o processo produtivo[7]. Por outro lado, a IoT de banda estreita (*Narrowband IoT*, NB-IoT) é uma tecnologia 4G de comunicação sem fios, normalizada pelo Projeto de Parceria de 3ª Geração (3GPP), desenvolvida especificamente para comunicações máquina em grande quantidade *massive Machine-Type Communications* (mMTC) [8]. A NB IoT é uma tecnologia LPWAN que funciona praticamente em qualquer lugar, interligando dispositivos de maneira mais simples e eficiente em redes móveis já estabelecidas e lida com pequenas quantidades de dados bidirecionais pouco frequentes, com segurança e confiabilidade [9].

## **AGRADECIMENTOS**

Os autores agradecem ao Instituto de Telecomunicações, e à FCT/MCTES que financiou este trabalho através de fundos nacionais e quando aplicável cofinanciado por fundos comunitários no âmbito do projeto com referência UIDB/EEA/50008/2020 e identificador DOI <https://doi.org/10.54499/UIDB/50008/2020>.

## REFERÊNCIAS

1. SILVA, T. A. A. da. "Percepção de Risco, Produção de Energia Eólica e Pequenos Agricultores do Agreste Pernambucano". CECS, v. 11, n. 1, jul. 2023. Disponível em: <<https://doi.org/10.47456/cadecs.v11i1.41925>>. Acesso em: 09 set. 2024.
2. DIGI INTERNATIONAL INC. "Guia do usuário XBee/XBee-PRO S1 802.15.4 - Sobre os módulos RF XBee/XBee-PRO S1 802.15.4 (legado)". Disponível em: <<https://www.digi.com/resources/documentation/Digidocs/90000982/Default.htm>>. Acesso em: 09 set. 2024.
3. BLYNK. "Introduction". Disponível em: <<https://docs.blynk.io/en/>>. Acesso em: 09 set. 2024.
4. COELHO, P. "5G e Internet das Coisas". Lisboa: FCA, Lisboa, Portugal, 2022.
5. LIBELIUM. "XBee X-CTU tutorial". Disponível em: <<https://development.libelium.com/xbee-x-ctu-tutorial/>>. Acesso em: 09 set. 2024.
6. PUNGO, É. M. S. *Eficiência Energética das Redes de Sensores sem Fios para Agricultura de Precisão*. Dissertação de Mestrado, Universidade da Beira Interior, Covilhã, PT: 2023.
7. PENTTINEN, J. T. J., Ed. *5G Second Phase Explained: The 3GPP Release 16 Enhancements*. 1st ed. Hoboken, NJ: Wiley, 2021.
8. MUTEBA, K. F.; DJOUANI, K.; OLWAL, T. "5G NB-IoT: Design, considerations, solutions and challenges". *Procedia Computer Science*, v. 198, p. 86–93, 2022.
9. D. J. S. D. Henriques, *Implementação De Um Sistema De Rega Inteligente*, Dissertação de Mestrado, Instituto Politécnico de Tomar, Tomar, Portugal, 2019. Disponível em: [https://comun.rcaap.pt/bitstream/10400.26/31426/1/Relatorio\\_Projeto\\_Final\\_Mestrado.pdf](https://comun.rcaap.pt/bitstream/10400.26/31426/1/Relatorio_Projeto_Final_Mestrado.pdf).