

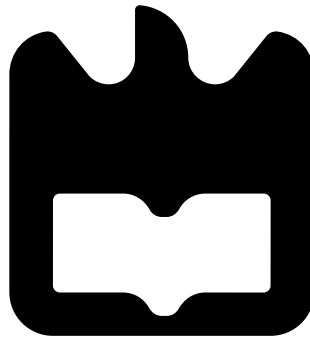


**Rui Pedro dos
Santos Oliveira**

**Sistema de monitorização e controlo da produção
de Salicórnica na Ria de Aveiro**

**Monitoring and control system for the production
of Salicornia in the ria of Aveiro**

DOCUMENTO PROVISÓRIO





**Rui Pedro dos
Santos Oliveira**

**Sistema de monitorização e controlo da produção
de Salicórnica na Ria de Aveiro**

**Monitoring and control system for the production
of Salicornia in the ria of Aveiro**

DOCUMENTO PROVISÓRIO



**Rui Pedro dos
Santos Oliveira**

**Sistema de monitorização e controlo da produção
de Salicórnica na Ria de Aveiro**

**Monitoring and control system for the production
of Salicornia in the ria of Aveiro**

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia de Computadores e Telemática, realizada sob a orientação científica do Doutor Joaquim Manuel Henriques de Sousa Pinto, Professor Associado do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro e do Doutor José Alberto Gouveia Fonseca, Professor Associado do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro.

o júri / the jury

presidente / president

ABC

Professor da Universidade de Aveiro

vogais / examiners committee

Doutor Joaquim Manuel Henriques de Sousa Pinto

Professor da Universidade de Aveiro (orientador)

GHI

Professor associado da Universidade J (co-orientador)

KLM

Professor Catedrático da Universidade N (arguente)

**agradecimientos /
acknowledgements**

palavras chave

cultivo da salicórnia, sensores, monitorização, atuadores, atuação remota, plataforma web, plataforma mobile, sistema de vídeo-vigilância

resumo

A evolução da Internet tem proporcionado o crescimento de novos modelos de negócio suportados por tecnologias de informação e comunicações e tem possibilitado o desenvolvimento de novos modelos de serviços baseados no cloud computing.

A exploração de novas descobertas na área da percepção visual, nomeadamente no que se refere à apreciação de obras de arte geniais, . . .

keywords

Cultivo da salicórnia, irrigação, sensores, atuadores, web, monitorização, atuação remota.

abstract

Nowadays, it is usual to evaluate a work . . .

Conteúdo

Lista de Figuras	iii
Lista de Tabelas	v
Acrónimos	vii
1 Estado da arte	1
1.1 Microcontroladores	2
1.1.1 Arduino Nano	2
1.1.2 Raspberry Pi 3	3
1.2 Sensores	4
1.2.1 Sensor de temperatura	4
1.2.2 Sensor de luminosidade	5
1.2.3 Sensor de salinidade	6
1.3 Tecnologias de comunicação	6
1.3.1 Zigbee	6
1.3.2 Bluetooth (BLE)	6
1.3.3 Wi-Fi (IEEE 802.11)	7
1.3.4 Sigfox	7
1.3.5 Comparação entre tecnologias de comunicação	7
2 Sistema de controlo e monitorização: arquitetura e modelação	9
2.1 Descrição global do sistema	9
2.2 Componentes	10
2.2.1 <i>Sensor Module</i>	11
2.2.2 <i>Controller Module</i>	12
2.2.3 Simulação em <i>hardware</i>	13
Sensores utilizados	13
Comunicação	16

3	Implementação	19
3.1	Simulação em <i>hardware</i>	20
3.1.1	Arduino Nano	20
	Sensores	20
	Comunicação	21
3.1.2	Raspberry Pi 3	21
3.1.3	Considerações finais	22
A	Interligação de componentes	29

Lista de Figuras

1.1	Arduino Nano	2
1.2	Identificação dos pinos num Arduino Nano	2
1.3	Raspberry Pi 3	3
1.4	Principais componentes no Raspberry Pi 3	3
1.5	Sensibilidade luminosa das plantas durante a fotossíntese)	5
2.1	Ilustração dos principais componentes do sistema	9
2.2	Ilustração da distribuição dos módulos em duas leiras	10
2.3	Esquema de componentes e respetiva comunicação entre três <i>Sensor Module</i> (SM) e um <i>Controller Module</i> (CM)	11
2.4	Sensor TTC 104 NTC	13
2.5	Esquema eletrotécnico da ligação do sensor de temperatura	13
2.6	Sensor foto-resistência GL5528	14
2.7	Esquema eletrotécnico da ligação do sensor de luminosidade	14
2.8	<i>Water Level Switch Liquid Level Sensor Plastic Ball Float</i>	15
2.9	Esquema eletrotécnico da ligação do sensor de nível líquido	15
2.10	Light Emitting Diode (LED)	15
2.11	Esquema eletrotécnico da ligação do LED	15
2.12	Comunicação entre componentes da simulação em <i>hardware</i>	16
2.13	Módulo Bluetooth HC-06	18
2.14	Esquema eletrotécnico da ligação do módulo Bluetooth	18
A.1	Protótipo de montagem de componentes eletrotécnicos	29

Lista de Tabelas

1.1	Características do Arduino Nano	3
1.2	Comparação entre versão 2 e 3 do Raspberry Pi	4
1.3	Comparação entre tecnologias de comunicação	7
2.1	Características do sensor TTC 104	14
2.2	Características do sensor GL5528	14
2.3	Características do módulo bluetooth HC-06	18

Acrónimos

API	Application Programming Interface	SM	<i>Sensor Module</i>
CM	<i>Controller Module</i>	CPU	Central Processing Unit
GPS	Global Positioning System	RAM	Random Access Memory
HTTP	HyperText Transfer Protocol	ISM	Industrial, Scientific, Medical
I/O	Input/ Output	LED	Light Emitting Diode
IDE	Integrated Development Environment	IP	Internet Protocol
IoT	<i>Internet of Things</i>	EDR	Enhanced Data Rate
LDR	Light Dependent Resistor	TCP	Transmission Control Protocol
NTC	Negative Temperature Coefficient	PAR	Photosynthetically Active Radiation
REST	Representational State Transfer	ROM	Read-Only Memory
REST	Representational State Transfer	USB	Universal Serial Bus
		PANs	Wireless personal area networks

Estado da arte

Neste capítulo, são apresentados os resultados da pesquisa bibliográfica sobre as ferramentas e funcionalidades que poderão estar presentes no sistema a desenvolver. Apresenta-se de forma geral todas as tecnologias possíveis de utilização e respetiva comparação.

1.1 Microcontroladores

Um microcontrolador consiste numa solução integrada de um sistema computacional, num único dispositivo físico, sendo uma mistura de *hardware* com *software*. Possui vários módulos principais, um Central Processing Unit (CPU) onde é realizado todo o processamento, as memórias de instrução e dados (Random Access Memory (RAM) e Read-Only Memory (ROM)) e os portos de entrada e saída (Input/ Output (I/O)), e ainda alguns periféricos (*Timer*, *Serial COM Port*).

Estes dispositivos são responsáveis pelo processamento de dados, tomada de decisões, controlo do funcionamento de alguns módulos, visualização e conversão da informação recolhida pelos sensores, entre outros. Com o objetivo de simular este sistema, serão utilizados dois microcontroladores bastante comuns no mercado, um Arduino Nano e um Raspberry Pi 3.

1.1.1 Arduino Nano

Como descrito no site oficial, um Arduino consiste numa plataforma *open-source* de prototipagem eletrónica composta por *hardware* e *software* flexíveis e com elevada facilidade de utilização. Esta plataforma é utilizada para projetos principalmente no contexto do *Internet of Things* (IoT) e da robótica educativa, sendo possível incorporar vários módulos, dependendo da tarefa que se quer executar[29].

O Arduino Nano possui um conjunto de pinos que podem ser programados para funcionarem como entradas ou saídas fazendo com que este microcontrolador interaja com o exterior para os mais diversos fins. Para além dos pinos de I/O existem pinos de alimentação que fornecem diversos valores de tensão que podem ser utilizados para transmitir energia elétrica aos diferentes componentes de um projeto[29]. Nas figuras 1.1 e 1.2 apresenta-se o Arduino utilizado e a identificação dos diferentes pinos existentes, respectivamente. Adicionalmente, na tabela 1.1 encontram-se as principais características desta versão do Arduino.

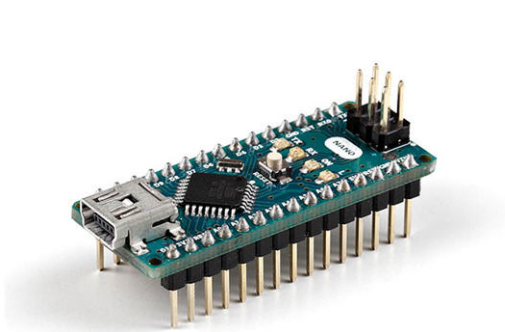


Figura 1.1: Arduino Nano

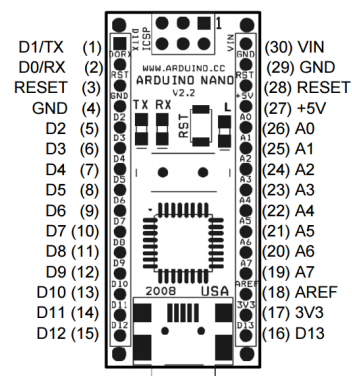


Figura 1.2: Identificação dos pinos num Arduino Nano (Retirado de [30])

Microcontrolador	ATmega328
Tensão de operação	5V
Tensão de entrada	7-12V
Portas digitais	14 (6 podem ser usadas como PWM)
Portas analógicas	8
Corrente nos pinos I/O	40mA
Memória Flash	32KB (2KB usado no bootloader)
Memória RAM (SRAM)	2KB
EEPROM	1KB
Velocidade do <i>Clock</i>	16MHz
Dimensões	45 x 18mm
LED interno	Pino digital 13
Ligação Universal Serial Bus (USB)	Ligação ao computador e alimentação

Tabela 1.1: Características do Arduino Nano (Adaptado de [31])

1.1.2 Raspberry Pi 3

Um Raspberry Pi (figura 1.3) é mais poderoso do que outros dispositivos de igual dimensão e incorpora alguns periféricos I/O, tal como um computador normal. Tem o tamanho de um cartão de crédito que possui um conjunto de *hardware* integrado que tal como o Arduino, possibilita uma interação com o exterior. Este dispositivo, desenvolvido no Reino Unido pela *Raspberry Pi Foundation*, pode ser ligado a um monitor através da saída HDMI, possuindo também uma saída de áudio e várias portas USB. O Raspberry Pi é compatível com sistemas operativos baseados em GNU/Linux, sendo que no trabalho prático desta dissertação será utilizada a versão 3 do Raspberry Pi com a versão Raspbian¹ instalada[32]..

Na figura 1.4 encontram-se os principais componentes da versão utilizada e na tabela 1.2 é feita um comparação entre a versão 2 e 3 do Raspberry Pi, permitindo concluir que a versão utilizada é mais poderosa.

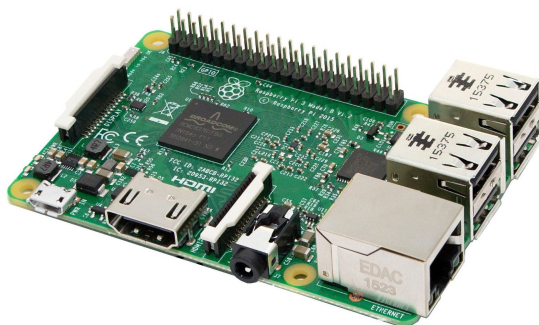


Figura 1.3: Raspberry Pi 3

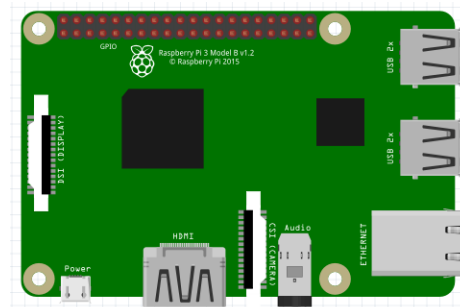


Figura 1.4: Principais componentes no Raspberry Pi 3

¹Distribuição Linux oficial do Raspberry Pi: <https://www.raspberrypi.org/downloads/raspbian/>

	Raspberry Pi 2 Model B 1.2	Raspberry Pi 3 Model B
CPU	QUAD Core 900MHz	QUAD Core 1.2GHz
RAM	1GB SDRAM	1GB SDRAM
Potência máxima	5V/1.8A	5V/2.5A
Armazenamento	MicroSD	MicroSD
USB 2.0	4 x Portas USB	4 x Portas USB
GPIO	40 pinos	40 pinos
Porta Ethernet	Sim	Sim
Wifi	Incorporado (versão 802.11n)	Não
Bluetooth LE	Incorporado (versão 4.1)	Não

Tabela 1.2: Comparação entre versão 2 e 3 do Raspberry Pi

1.2 Sensores

Esta secção tem como objetivo fazer um estudo comparativo entre as diferentes tecnologias usadas para a medição dos vários parâmetros ambientais necessários ao controlo e monitorização da produção de Salicórnica. Serão descritos alguns sensores de temperatura, luminosidade e salinidade existentes no mercado.

1.2.1 Sensor de temperatura

Existem vários tipos de sensores de temperatura baseados em princípios de funcionamento distintos, nomeadamente os termopares, os termístores e os de circuito integrado.

- **Termopares:** são sensores de temperatura simples, robustos e de baixo custo, constituídos por duas partes diferentes de material condutor. Podem medir temperaturas entre os -200°C e os 2315°C , sendo usados em grande escala[33].
- **Termístor:** são sensores cuja resistência varia com a temperatura, sendo construídos a partir de materiais semicondutores. Um termístor detém uma maior sensibilidade, pois uma pequena variação de temperatura provoca uma grande variação na sua resistência, permitindo que a temperatura típica seja entre os -100°C e os 300°C . Estes sensores são frágeis, baratos e de dimensões reduzidas, sendo suscetíveis a problemas de auto aquecimento[34].
- **Circuito integrado:** são sensores construídos através de materiais semicondutores, o que possibilita que tenham uma gama de temperatura limitada, geralmente entre os -55°C e os 150°C . Estão disponíveis com saídas em tensão, ou corrente, linearmente proporcional à temperatura.

1.2.2 Sensor de luminosidade

Existe uma enorme variedade de sensores de luminosidade/radiação no mercado, contudo para monitorizar a luminosidade incidente numa planta é fundamental conhecer a radiação que é utilizada no processo de fotossíntese, denominada por Photosynthetically Active Radiation (PAR). A figura 1.5 ilustra o espectro de absorção de luz pelas plantas. É possível ver que estas são mais sensíveis à luz azul e vermelha (300-700 nm).

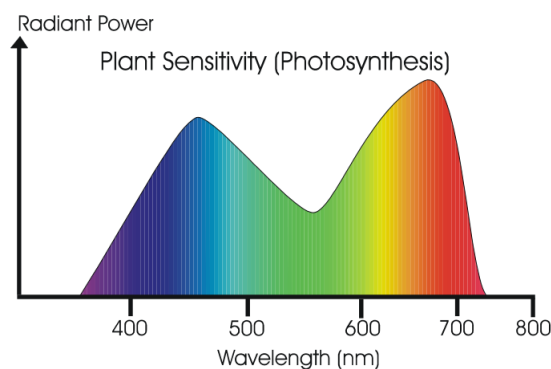


Figura 1.5: Sensibilidade luminosa nas plantas durante a fotossíntese (Retirado de [35])

Alguns dos sensores de luminosidade/radiação mais comuns no mercado são os sensores PAR, os piranómetros e os sensores optoelectrónicos, descritos de seguida.

- **Sensores PAR:** são desenhados especificamente para medir a PAR, sendo indicados para medir a luz incidente numa planta. Os custos associados a este tipo de sensores são elevados e por isso são utilizados maioritariamente na investigação hortícola.
- **Piranómetros:** estes sensores medem a radiação solar total (radiação ultravioleta, visível e infravermelha), devendo ser utilizados no exterior. Este tipo de sensores é particularmente útil em estações meteorológicas. No entanto, são bastante dispendiosos, podendo ser mais caros dos que os sensores PAR.
- **Sensores optoelectrónicos:** são dispositivos eletrónicos que se baseiam em fenómenos fotovoltaicos. Existem três sensores deste tipo, que possuem um elevado desempenho quando afetados pelo ruído e uma elevada sensibilidade. São considerados os sensores de luminosidade mais económicos.
 - **Fotodíodo:** pode ser visto como um díodo comum, gerando uma corrente ou tensão quando incide radiação sobre ele.
 - **Fototransístor:** é um transístor desenhado para receber luz, normalmente num módulo transparente.

- **Light Dependent Resistor (LDR)**: trata-se de um material semicondutor cuja resistência diminui com a incidência de luz. Trata-se de um sensor económico e de fácil utilização.

1.2.3 Sensor de salinidade

Atualmente ainda não há disponível nenhum sensor que permite medir exatamente a salinidade. Contudo, existem sensores que medem a condutividade elétrica num determinado meio, permitindo concluir o estado do meio relativamente à quantidade de sal existente. Um exemplo disso, é o sensor SAL-BTA da Vernier[36] que permite medir de forma fácil e precisa o teor de sal dissolvido numa solução aquosa.

1.3 Tecnologias de comunicação

Nesta secção serão apresentadas algumas das tecnologias de comunicação sem fios mais utilizados em *Internet of Things* que permitem a troca de informação entre dispositivos. Serão abordados quatro tecnologias bastante populares comunicação: Zigbee, Bluetooth, Wi-Fi e SigFox. Por fim, é realizada uma comparação entre as tecnologias estudadas.

1.3.1 Zigbee

Zigbee é um padrão de rede sem fios destinado a aplicações de controlo e sensores remotos, adequado para operações em ambientes isolados. Esta tecnologia é de curto alcance, baixa complexidade e consumo de energia, possuindo ainda uma taxa de dados baixa. Permite uma topologia de rede *mesh* (malha) permitindo elevados níveis de fiabilidade e maior alcance de cobertura, fornecendo mais do que um caminho através da rede para qualquer ligação[37].

1.3.2 Bluetooth (BLE)

Bluetooth (BLE) é uma marca característica da versão 4.0 do Bluetooth, sendo projetado para aplicações de baixa potência, utilizada em comunicações de curta distância. Esta tecnologia de comunicação bastante conhecida, consiste numa especificação de rede sem fio de âmbito pessoal, também conhecida por Wireless personal area networks (PANs), sendo de baixo custo e consumo de energia. O Bluetooth surgiu com a intenção de eliminar o elevado número de cabos existentes na comunicação entre dispositivos, sendo que este padrão funciona na banda de frequência 2,4GHz, conhecida como banda Industrial, Scientific, Medical (ISM)², disponível mundialmente sem a necessidade de licenças[38][39].

²Faixa de frequência compartilhada por dispositivos, como forno micro-ondas e Wi-Fi

1.3.3 Wi-Fi (IEEE 802.11)

O Wi-Fi é uma tecnologia de recepção de transmissão de dados sem fios, baseado no *standard* IEEE 802.11. Com o evoluir da tecnologia, têm surgido diferentes variações do mesmo *standard*, sendo a variante 802.11g a mais comum. Seguidamente, são apresentadas as características principais desta variante[40][41]:

- Banda de frequência de 2.4 GHz;
- Taxa de bits máxima de 54 Mbps na camada física, transferência de dados máxima de 24.7 Mbps;
- Topologia em estrela (infraestrutura) e ponto-a-ponto (ad-hoc);
- Vários mecanismos de encriptação e autenticação;

1.3.4 Sigfox

A tecnologia Sigfox permite a comunicação sem fios entre dispositivos possuindo um baixo consumo de energia, utilizando a banda ISM, tal como o Bluetooth. De acordo com o site oficial, o Sigfox é considerado um protocolo leve, permitindo trocas de pequenas mensagens possibilitando um consumo de bateria bastante reduzido[42]. Esta tecnologia é totalmente virada para o IoT devido ao seu baixo consumo.

1.3.5 Comparação entre tecnologias de comunicação

Na tabela 1.3 é apresentada uma comparação entre algumas características das tecnologias anteriormente estudadas.

De acordo com os dados apresentados na tabela, concluímos que o Zigbee e o Sigfox são tecnologias de baixo consumo, contudo o alcance destas é inferior relativamente às tecnologias Bluetooth e Wi-Fi.

	Zigbee	Bluetooth	Wi-fi	Sigfox
IEEE	802.15.4	802.15.1	802.11n	N/A
Banda de frequência	868/915 MHz 2.4 GHz	2,4-5,5 GHz	2,4-5 GHz	800 Hz (europa)
Topologia da rede	Mesh	Estrela	Estrela	P2P
Energia (consumo)	Muito baixo	Baixo	Médio	Muito baixo
Alcance	10m	aprox. 50m	100m	Rural: 10-15m Urbano: 3-5m
Bateria (duração)	Meses-anos	Dias	Horas	10/15 anos

Tabela 1.3: Comparação entre tecnologias de comunicação (Adaptado de [37])

Sistema de controlo e monitorização: arquitetura e modelação

Este capítulo tem como principal objetivo a descrição do sistema que resultou do trabalho prático desta dissertação. Cada elemento do sistema é caracterizado de acordo com as suas funções, especificidades e arquitetura, bem como a forma como os elementos interagem entre si. Para além disso, é apresentado todo o processo de modelação do sistema tendo por base os requisitos do cliente.

2.1 Descrição global do sistema

Este sistema tem como objetivo a supervisão remota da produção de Salicórnia, permitindo não só a monitorização dos dados adquiridos pelos sensores, como também da atuação remota de determinados comandos. Neste contexto, também é possível a aquisição de imagens que possibilita a deteção de intrusos nas quintas onde se realiza a produção desta espécie. O esquema da figura 2.1 ilustra de um modo geral todos os componentes e as diferentes plataformas com que o cliente pode interagir.

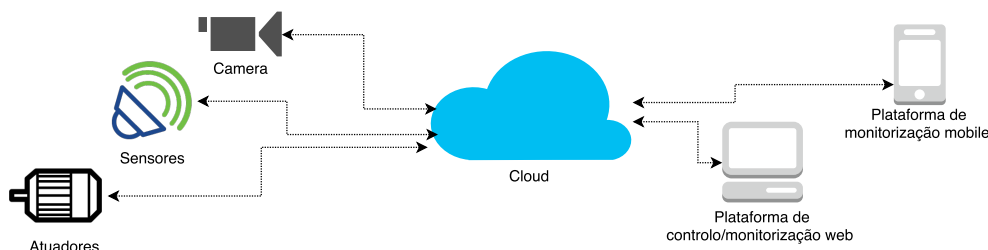


Figura 2.1: Ilustração dos principais componentes do sistema

Como vimos no capítulo 3, uma plantação de Salicórnia carece de um controlo relativa-

mente fino de certos parâmetros ambientais sobretudo da salinidade do terreno, que depende, das chuvas, da salinidade da água dos canais da ria, entre outros. Nas quintas onde se cultiva Salicórnia, a produção faz-se numa espécie de leiras limitadas por pequenos canais de irrigação que podem ser cheios de água salgada proveniente dos esteiros que rodeiam a quinta. Esta operação implica a abertura de válvulas de admissão de água, medida do nível da maré nos canais, monitorização da qualidade e salinidade da água exterior.

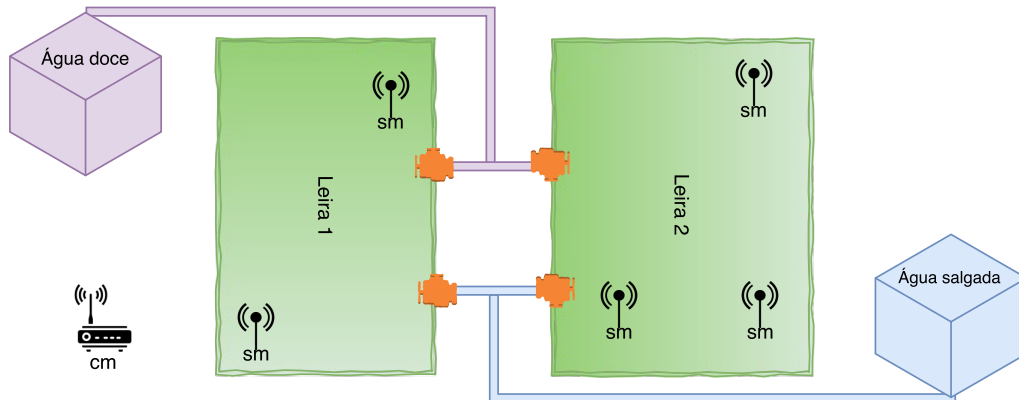


Figura 2.2: Ilustração da distribuição dos módulos em duas leiras

Tal como ilustrado na figura 2.2, foram colocados módulos com sensores distribuídos estrategicamente por cada leira. Cada um desses módulos, irá comunicar com um módulo central originando uma topologia de rede em estrela. Por sua vez, cada um destes módulos centrais irá comunicar diretamente com o servidor que receberá todos os dados adquiridos tanto pelos sensores como por atuadores, permitindo que estes sejam guardados numa base de dados específica. Os atuadores, permitirão despoletar ações que autorizam a ativação ou desativação de bombas e/ou válvulas para transferências de águas de modo a melhorar as condições de cultivo da Salicórnia. Os módulos centrais têm acesso à camada protocolar TCP/IP (Internet) de modo a conseguirem a utilização da API REST via HyperText Transfer Protocol (HTTP) desenvolvida para o efeito.

No que diz respeito às plataformas de interação com o cliente, existe uma *dashboard* e uma aplicação *mobile*. A *dashboard* disponibiliza uma interface que apresenta as informações mais importantes para o utilizador de forma apelativa, tornando mais fácil a sua interação e respetiva leitura, possibilitando ainda a gestão de todo o sistema e realização de operações de controlo remoto. Por outro lado, a aplicação *mobile* permite apenas a monitorização do cultivo da Salicórnia e receção de alertas quando estes ocorrem.

2.2 Componentes

No contexto desta dissertação é necessário reter dois conceitos principais, são eles:

- **Sensor Module:** consiste num módulo responsável pela aquisição de dados provenientes dos mais diversos tipos de sensores.
- **Controller Module:** consiste num módulo responsável pela receção dos dados/estados do *Sensor Module* e respetivo envio para a *cloud*.

O cenário da figura 2.3 ilustra três *Sensor Modules* que comunicam com um *Controller Module*. Cada um desses *Sensor Module* possui um conjunto específico de sensores, podendo estes ser atuadores ou câmaras. Para a comunicação com o *Controller Module*, cada *Sensor Module* possui um determinado módulo de comunicação que permite a transferência dos dados adquiridos pelos sensores. Posteriormente, o *Controller Module* possui um determinado protocolo de comunicação (TPC/IP) que permite a utilização da Application Programming Interface (API) e respetivo envio ou atualização dos dados adquiridos pelo sistema.

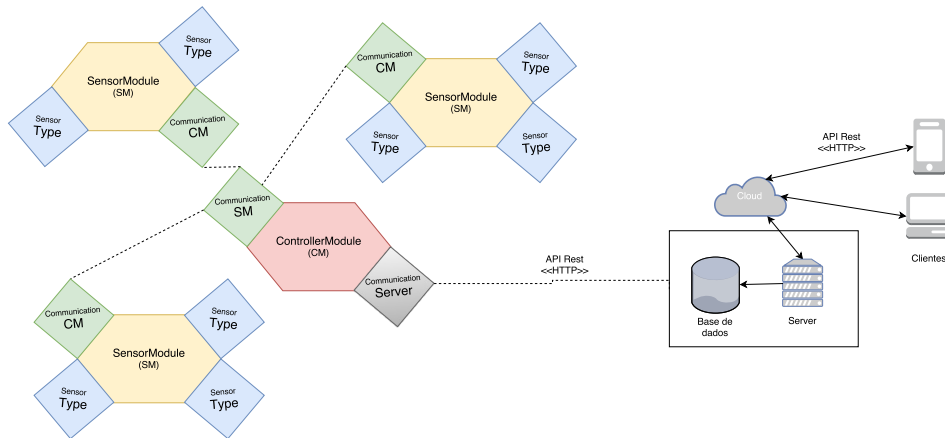


Figura 2.3: Esquema de componentes e respetiva comunicação entre três SM e um CM

Seguidamente serão especificados todos os detalhes de cada módulo, uma vez que serão considerados na modelação de todo o sistema.

2.2.1 Sensor Module

Um *Sensor Module* consiste num micro-controlador responsável pela aquisição de dados provenientes dos mais diversos tipos de sensores, podendo estes ser atuadores ou câmaras. No caso de se tratar de um atuador, isto é, válvulas, bombas, contadores, pás ou cancelas, apenas serão lidos valores binários. Caso se trate de uma câmara, todo o processamento é feito internamente nesta, sendo que o sistema apenas irá receber o IP da mesma.

Tal como referido anteriormente, cada *Sensor Module* terá que utilizar um determinado módulo de comunicação para possibilitar a transferência dos dados adquiridos para o módulo central. Para além disso, pretende-se que o *Sensor Module* em condições extremas, possa tomar decisões de atuação, isto é, caso seja lido um valor fora do padrão e que seja necessário a

ativação de um atuador, este deverá ser auto suficiente em tomar esta decisão, sem necessidade de intervenção do utilizador.

Pretende-se que este módulo seja identificado por um determinado nome, possua uma bateria que permita a sua mobilidade, tenha um ou vários módulos de comunicação acoplados que permitam comunicar com um módulo central, uma memória e um módulo Global Positioning System (GPS) que permita aceder à sua localização, identificando-o em caso de furto. Para além disso, um *Sensor Module* terá que possuir obrigatoriamente um ou vários sensores.

2.2.2 *Controller Module*

Um *Controller Module* consiste num micro-controlador responsável pela receção dos dados provenientes dos vários *Sensor Modules*. Pretende-se que este módulo envie ou receba informações para os *Sensor Module* quando solicitados pelo utilizador. Após a receção dos dados, estes são enviados para um servidor em *cloud* através de uma API Representational State Transfer (REST) criada para o efeito. Sendo que a tecnologia REST opera sob o protocolo de comunicação HTTP, este componente tem que necessariamente estar ligado à rede Internet via Transmission Control Protocol (TCP)/Internet Protocol (IP).

É essencial que este módulo possua alguma capacidade de processamento, uma vez que poderá ter vários *Sensor Modules* a si associados e com necessidade de constante envio e receção de dados. Para além disso, pretende-se que o *Controller Module* seja identificado por um determinado nome, tenha um módulo de comunicação que possibilite o envio de dados para um servidor e outros para comunicação com os diferentes *Sensor Modules*. Tal como acontece com os *Sensor Modules*, existe necessidade de acoplado um módulo GPS que permita localizar o micro-controlador em caso de robo.

2.2.3 Simulação em *hardware*

Após a desenvolvimento da API, simulou-se o sistema num contexto real. Para tal, pretendia-se encontrar *hardware* que encaixasse no contexto deste projeto. Foram utilizados dois microcontroladores (Arduino Nano e Raspberry Pi 3) e alguns sensores. Para este cenário, assume-se que o Arduino Nano é considerado um *Sensor Module* que possui um conjunto de sensores enquanto que o Raspberry Pi 3 é um *Controller Module* que recebe os dados provenientes do *Sensor Module* enviando-os para o servidor.

Seguidamente, são apresentados os sensores utilizados bem como os tipos de comunicação.

Sensores utilizados

Nesta secção apresentam-se os sensores utilizados na simulação e as suas principais características. Todos os sensores foram escolhidos tendo em conta o seu enquadramento no projeto e a sua disponibilidade em laboratório, sendo que serão ligados ao Arduino Nano.

Temperatura

Como sensor de temperatura foi utilizado um termistor do tipo Negative Temperature Coefficient (NTC). Como vimos no capítulo 1, um termistor é um semicondutor sensível à temperatura, ou seja, quando o coeficiente de variação da resistência com a temperatura é negativa, então a temperatura sobe e consequentemente a resistência diminui. Na figura 2.5 encontra-se o esquema de ligação deste componente e na tabela 2.1 as suas propriedades principais[52].



Figura 2.4: Sensor TTC 104 NTC

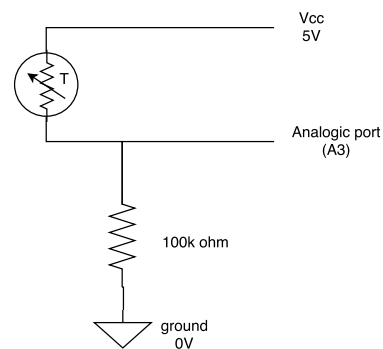


Figura 2.5: Esquema eletrotécnico da ligação do sensor de temperatura

Dimensão	5mm
Resistência	100 K Ω
Valor máximo	+125 °C
Valor mínimo	-30 °C
Nível de confiança	$\pm 10\%$
Preço	0.35 e/unidade

Tabela 2.1: Características do sensor TTC 104 (Adaptado de [52])

Luminosidade

Para simular a luminosidade incidente foi utilizado um sensor do tipo foto-resistência, neste caso o GL5528 (figura 2.6). Este sensor, também conhecido como LDR, não é mais do que uma resistência variável cujo o seu valor varia conforme a intensidade da luz que incide sobre ele, isto é, à medida que a intensidade da luz aumenta, a sua resistência diminui. Este sensor tem múltiplas aplicações, entre as quais se destaca a monitorização solar, indicador da posição do sol (*up/down*), alarmes anti-roubo, alarme para abertura/fecho de portas entre outras. Como vimos no capítulo 1 é um sensor de baixo custo e de fácil utilização. Na figura 2.7 encontra-se o esquema de ligação do componente e na tabela 2.2 são apresentadas as principais características do sensor utilizado.

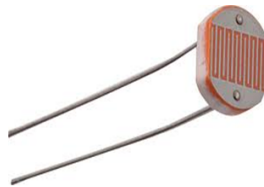


Figura 2.6: Sensor foto-resistência GL5528

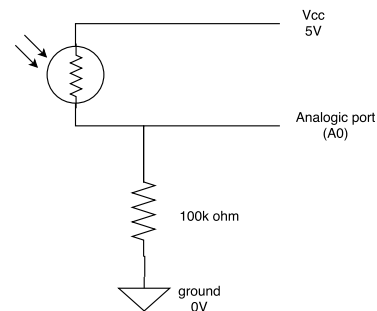


Figura 2.7: Esquema eletrotécnico da ligação do sensor de luminosidade

Diâmetro	5 mm
Tensão máxima	150 VDC
Potência máxima	100 mW
Tensão de operação	-30 °C a 70 °C
Espectro	540 nm
Comprimento com terminais	32 mm
Resistência na luz	10-20 K Ω (Lux 10)
Material	Carbono
Preço	0.22 €/unidade

Tabela 2.2: Características do sensor GL5528 (Adaptado de [53])

Sensor para verificação do estado do nível de água

Este sensor, denominado por *Water Level Switch Liquid Level Sensor Plastic Ball Float* (figura 2.8), não é mais do que um interruptor que é ativo sempre que um determinado líquido ultrapassa o mesmo, isto é, sempre que algum líquido atingir o pedaço de plástico este irá subir ativando assim o circuito. Na figura 2.9 encontra-se o esquema da ligação deste sensor.

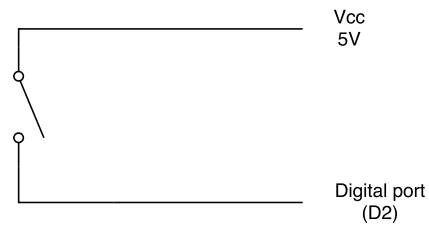


Figura 2.8: *Water Level Switch Liquid Level Sensor Plastic Ball Float*

Figura 2.9: Esquema eletrotécnico da ligação do sensor de nível líquido

Simulador de válvula para transferências de águas

Para a simulação de uma válvula que permitirá as transferência de água doce e/ou água salgada foi utilizado um LED. Este possibilita facilmente identificar através da sua ativação se a válvula se encontra ativa ou não. Na figura 2.11 encontra-se o esquema de ligação deste componente.

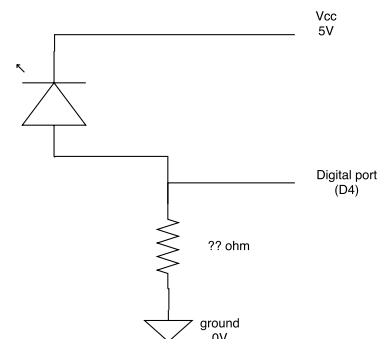


Figura 2.10: LED

Figura 2.11: Esquema eletrotécnico da ligação do LED

Comunicação

Nesta secção, apresentam-se os tipos de comunicação para o cenário escolhido. Pretendia-se que cada um dos módulos fique isolado, o que implicou o estudo e respetiva escolha de algumas tecnologias de comunicações sem fios (secção 1.3 do capítulo 1).

De acordo com o verificado na tabela 1.3, o Zigbee e o Sigfox apresentam características que melhor se adaptam ao conceito de IoT. No entanto, para o contexto deste trabalho, privilegia-se o alcance das tecnologias Bluetooth e Wifi dado que, pretende-se tirar partido desta característica e distribuir os *Sensor Modules* pela máxima distancia possível. Para além disso, estas eram as duas únicas tecnologias disponíveis no laboratório.

- **Bluetooth:** utilizado para a comunicação entre o Arduino Nano e o Raspberry Pi 3. No Arduino, foi utilizado um módulo Bluetooth HC-06 e no caso do Raspberry Pi 3 foi utilizado o seu módulo interno (versão 4.1).
- **Wi-Fi:** utilizado para a comunicação entre o Raspberry Pi 3 e o servidor web, sendo utilizado o seu módulo interno (802.11.g).

O esquema da figura 2.12 ilustra os tipos de comunicação envolvidos nesta simulação para cada um dos componentes.

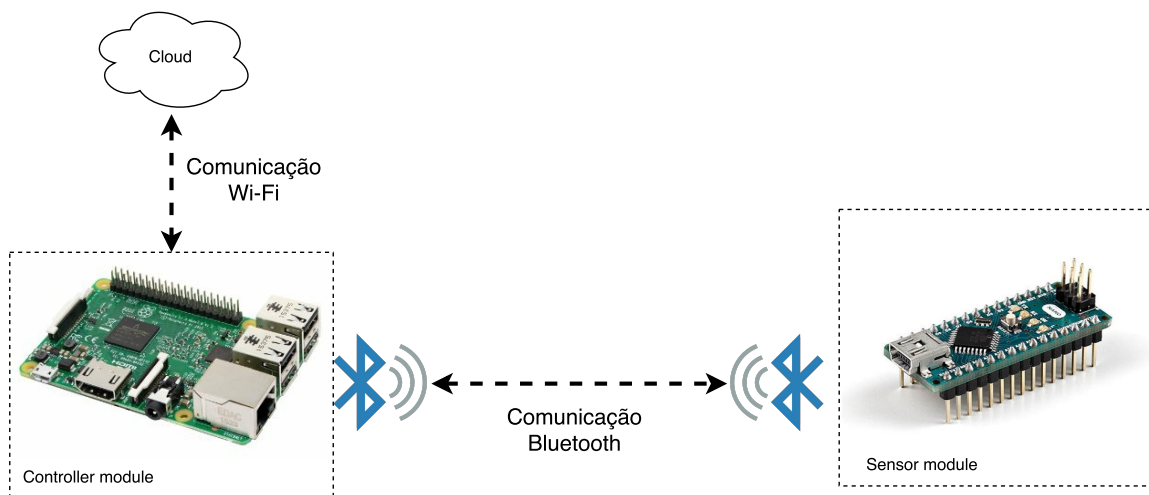


Figura 2.12: Comunicação entre componentes da simulação em *hardware*

Módulo Bluetooth HC-06

Este módulo Bluetooth oferece uma forma simples de envio e receção de informações remotamente, podendo ser adquirido a um custo reduzido. Este componente funciona apenas em modo *slave*, isto é, apenas permite que outros dispositivos se liguem a si, mas não permite que ele próprio se ligue a outros. Para além disso, possui um LED que permite indicar se algum dispositivo está emparelhado. É um dos módulos Bluetooth mais comuns no microcontrolador Arduino, possuindo um alcance máximo de aproximadamente 10 metros[54].

Na figura 2.14 encontra-se o esquema de ligação deste módulo e na tabela 2.3 são apresentadas as suas principais características.

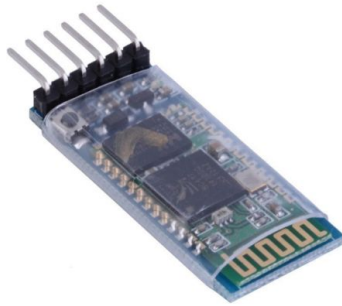


Figura 2.13: Módulo Bluetooth HC-06

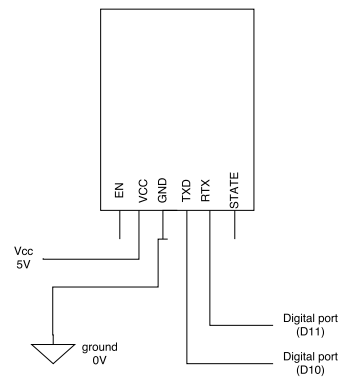


Figura 2.14: Esquema eletrotécnico da ligação do módulo Bluetooth

Versão Bluetooth	v2.0 com Enhanced Data Rate (EDR)
Frequência	2,4GHz Banda ISM
Segurança	Autenticação (PIN) e Encriptação
Tensão	Aconselhada 3,3v (2,7v - 4.2v)
Alcance	10 metros
Dimensões	26,9 x 13 x 2,2mm
Peso	9,6g
Temperatura (funcionamento)	-25C +75C
Preço	5.26 €/unidade

Tabela 2.3: Características do módulo bluetooth HC-06 (Adaptado de [54])

3

Implementação

3.1 Simulação em *hardware*

Nesta secção explica-se a implementação a nível de *software* no contexto desta simulação para cada um dos microcontroladores.

3.1.1 Arduino Nano

No que diz respeito ao Arduino Nano (*Sensor Module*), numa fase inicial, procedeu-se à ligação dos diversos componentes apresentados na secção 2.2.3 a uma placa branca (*breadboard*) tal como se apresentada no Anexo A. Para auxiliar o desenvolvimento de *software* foi utilizada a versão 1.8.1 do Integrated Development Environment (IDE) do próprio Arduino¹. Seguidamente apresenta-se a implementação necessária a nível de sensores e de comunicação.

Sensores

Foram desenvolvidos os seguintes métodos que permitem aceder aos valores lidos de cada um dos sensores. Para além disso, foi criado um método que permite alterar o estado de ativação do LED, permitindo simular o estado da válvula para transferência de águas.

- `int readTemperature(int port)`: é efetuada uma leitura no porto analógico e seguidamente realizada uma conversão para °C (graus Celsius);
- `long readLuminosity(int port)`: é efetuada uma leitura ao porto analógico e posteriormente é realizada uma conversão para percentagem (%);
- `int readWaterValve(int port)`: é efetuada uma leitura no porto digital através do método `digitalRead` disponibilizado pelo Arduino.
- `int readWaterLevel(int port)`: é realizada uma leitura no porto digital através do método `digitalRead`.
- `void setWaterValve(int port, int state)`: se a variável `state` for 1 então o porto é colocado a HIGH (1) através do método `digitalWrite`, caso contrário é colocado a LOW (0)

Inicialmente procedeu-se à leitura de cada sensor de forma individual de modo a garantir o seu total funcionamento. Sempre que é enviado um pedido de leitura dos sensores pelo *Controller Module* os valores são enviados no formato apresentado em 3.1.

$$\langle \text{temperatura} \rangle; \langle \text{nível_água} \rangle; \langle \text{luminosidade} \rangle; \langle \text{estado_válvula} \rangle \quad (3.1)$$

¹<https://www.arduino.cc/en/Main/Software>

Comunicação

Numa primeira fase, procedeu-se à comunicação entre o *Sensor Module* e *Controller Module* através de porta série. Seguidamente, optou-se por incorporar o módulo Bluetooth de modo a tornar os dois módulos independentes. Foi utilizado o *package SoftwareSerial.h* disponibilizado pelo Arduino, que permite interagir facilmente com o módulo de comunicação utilizado. Depois, decidiu-se quais os *inputs* que o Arduino irá receber e que ações iria executar, concluindo-se que podiam ser recebidos valores entre 0 e 2.

- 0: ativação do LED (válvula para transferência de água)
- 1: desativação do LED (válvula para transferência de água)
- 2: requisitar dados obtidos pelos sensores existentes no formato definido em 3.1

Antes de proceder à implementação de envio e receção de dados por Bluetooth no Raspberry Pi 3, testou-se esta funcionalidade isoladamente. Para isso, utilizou-se uma aplicação existente na *Play Store* denominada de *Bluetooth Terminal HC-05*², que permitiu facilmente validar este mecanismo. Os resultados deste teste funcional serão apresentados no próximo capítulo.

3.1.2 Raspberry Pi 3

Nesta simulação, o *Controller Module* recebe apenas os dados adquiridos por um *Sensor Module*, e envia as ações que este terá que executar. Para a comunicação entre os dois módulos, foi utilizado o módulo interno Bluetooth 4.1 disponível no hardware do Raspberry Pi 3.

De modo a conseguir receber os dados adquiridos e enviá-los para o servidor através da API, desenvolveu-se um *script* em Python que permite o seguinte:

1. Verificação dos dispositivos Bluetooth disponíveis
2. Estabelecer uma ligação com módulo HC-06 através de um *socket* de comunicação. Para tal, foi utilizado o *package socket* disponibilizado pelo Python.
3. Aceder à API de modo a verificar o estado LED (válvula de admissão) e enviá-lo para o *Sensor Module* através do *socket* utilizando o método `send()`.
4. No caso de ser enviado o dígito '1' o LED será ligado (válvula aberta), enquanto que se for enviado o dígito '0' o LED será desligado (válvula fechada).
5. No caso de ser enviado o dígito '2', o *socket* ficará a aguardar a receção dos dados lidos pelos sensores no formato definido, utilizando para isso o método `recv()`.

²<https://play.google.com/store/apps/details?id=project.bluetoothterminal>

6. Após a recepção dos dados lidos, é efetuado algum processamento para que estes sejam enviados através da API para o servidor.
7. Existem dois *threads*: um que permite pedir e receber os dados lidos pelos sensores e outro que permite controlar o LED (válvula)
8. Os dados lidos pelos sensores são enviados com um atraso igual ao *seding time* definido pelo utilizador na *dashboard*.

Para permitir o acesso aos recursos do sistema Bluetooth foi utilizada uma extensão do Python denominada de *pybluez*³.

3.1.3 Considerações finais

Simulou-se o sistema recorrendo a *hardware* disponível em laboratório, tanto a nível de microcontroladores, sensores ou módulos de comunicação. No entanto, não houve oportunidade de testar nenhum sensor de salinidade apesar de ser bastante importante no contexto do projeto, uma vez que é um dos parâmetros principais necessários de monitorização no cultivo da Salicórnica.

Numa primeira fase, toda a comunicação foi realizada recorrendo ao método mais simples, por porta série. Todos os dados recebidos eram enviados para o servidor através da API criada para o efeito. Posteriormente, optou-se por isolar os módulos sem necessidade de cabos, utilizando para isso um módulo Bluetooth.

³<https://github.com/karulis/pybluez>

Bibliografia

- [1] João Silva, “Sal verde, National Geographic.” [Online]. Available: <https://nationalgeographic.sapo.pt/23-arquivo/as-nossas-historias/298-sal-verde> [Accessed: 2017-02-01]
- [2] S. Beer and O. Demina, “A new species of *Salicornia* (Chenopodiaceae) from European Russia,” pp. 253–257, 2005.
- [3] M. Ferri and N. Menezes, *Glossário Ilustrado de Botânica*, 1st ed., Livraria Nobel, Ed., Brasil, 1981.
- [4] M. H. A. Silva, “Aspectos morfológicos e ecofisiológicos de algumas halófitas do sapal da Ria de Aveiro,” Ph.D. dissertation, Universidade de Aveiro, 2000. [Online]. Available: <http://ria.ua.pt/handle/10773/925>
- [5] V. Isca, A. Seca, D. Pinto, and A. Silva, *An overview of Salicornia genus: the phytochemical and pharmacological profile*, natural pr ed., V. Gupta, Ed. Daya Publishing House, New Delhi, 2014.
- [6] E. Figueroa, J. Jimenez-Nieva, J. Carranza, and C. Gonzalez Vilches, “Distribucion y Nutricion Mineral de *Salicornia ramosissima* J. Woods, *Salicornia europaea* L. y *Salicornia dolichostachya* Moss. en el estuario de los rios Odiel y Tinto (Huelva, SO España),” *Limnetica*, vol. 3, no. 2, pp. 307–310, 1987.
- [7] R. Pinto, “Expresso — A planta que é uma alternativa ao sal: antes era uma praga, agora é uma erva gourmet,” 2015. [Online]. Available: <http://bit.ly/1PR7KAG> [Accessed: 2017-02-01]
- [8] A. J. Davy, G. F. Bishop, and C. S. B. Costa, “*Salicornia* L. (*Salicornia pusilla* J. Woods, *S. ramosissima* J. Woods, *S. europaea* L., *S. obscura* P.W. Ball & Tutin, *S. nitens* P.W. Ball & Tutin, *S. fragilis* P.W. Ball & Tutin and *S. dolichostachya* Moss),” *Journal of Ecology*, vol. 89, no. 4, pp. 681–707, 2001.
- [9] H. Silva, G. Caldeira, and H. Freitas, “*Salicornia ramosissima* population dynamics and tolerance of salinity,” *Ecological Research*, vol. 22, no. 1, pp. 125–134, 2007.

- [10] A. Rubio-Casal, J. Castillo, C. Luque, and M. Figueroa, "Influence of salinity on germination and seeds viability of two primary colonizers of Mediterranean salt pans," *Journal of Arid Environments*, vol. 53, no. 2, pp. 145–154, feb 2003. [Online]. Available: <http://linkinghub.elsevier.com/retrieve/pii/S0140196302910426>
- [11] M. Filomena, D. J. Raposo, R. Manuel, S. Costa, A. Maria, and M. Bernardo, "Controlled atmosphere storage for the preservation of *Salicornia ramosissima*," no. October 2016, 2009.
- [12] Y. Ventura, W. A. Wuddineh, M. Myrzabayeva, Z. Alikulov, I. Khozin-Goldberg, M. Shpigel, T. M. Samocha, and M. Sagi, "Effect of seawater concentration on the productivity and nutritional value of annual *Salicornia* and perennial *Sarcocornia* halophytes as leafy vegetable crops," *Scientia Horticulturae*, vol. 128, no. 3, pp. 189–196, apr 2011. [Online]. Available: <http://linkinghub.elsevier.com/retrieve/pii/S0304423811000537>
- [13] Q. Z. Wang, X. F. Liu, Y. Shan, F. Q. Guan, Y. Chen, X. Y. Wang, M. Wang, and X. Feng, "Two new nortriterpenoid saponins from *Salicornia bigelovii* Torr. and their cytotoxic activity," *Fitoterapia*, vol. 83, no. 4, pp. 742–749, jun 2012. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/22414316><http://linkinghub.elsevier.com/retrieve/pii/S0367326X12000640>
- [14] D. Evans, "A Internet das Coisas Como a próxima evolução da Internet está mudando tudo," pp. 5–7, 2011.
- [15] B. Getting, "Basic Definitions: Web 1.0, Web 2.0, Web 3.0 — Practical Ecommerce." [Online]. Available: <http://www.practicalecommerce.com/articles/464-Basic-Definitions-Web-1-0-Web-2-0-Web-3-0> [Accessed: 2017-02-20]
- [16] J. Lovato, "Google's evolution in 10 years," 2014. [Online]. Available: <http://www.mediavisioninteractive.com/blog/search-enginenews/looking-back-moving-forward-google-evolution/> [Accessed: 2017-02-20]
- [17] T. Our, "Resume : Context Aware Computing for The Internet of Things : A Survey Article 2013," pp. 1–5, 2013.
- [18] J. Rowley, "The wisdom hierarchy: representations of the DIKW hierarchy," *Journal of Information Science*, vol. 33, no. 2, pp. 163–180, 2007. [Online]. Available: <http://alturl.com/7qike>
- [19] MySQL, "MySQL :: About MySQL," 2011. [Online]. Available: <http://www.mysql.com/about/> [Accessed: 2017-07-13]

-
- [20] “SQL Server 2017 on Windows and Linux — Microsoft,” 2017. [Online]. Available: <https://www.microsoft.com/en-us/sql-server/sql-server-2017> [Accessed: 2017-07-17]
- [21] The PostgreSQL Global Development Group, “PostgreSQL: About,” 2012. [Online]. Available: <https://www.postgresql.org/about/http://www.postgresql.org/about/> [Accessed: 2017-05-29]
- [22] DB-Engines, “Historical Trend of the Popularity Ranking of Database Management Systems,” 2014. [Online]. Available: <http://db-engines.com/en/ranking{ }trend> [Accessed: 2017-07-13]
- [23] I. V. López and G. B. Gutiérrez, “El Benchmark TPC-H en MySQL y PostgreSQL,” *Ingenieria.Lm.Uasnet.Mx*, pp. 1–7, 2009. [Online]. Available: <http://ingenieria.lm.uasnet.mx/sitio/congreso/documentos/iso15.pdf>
- [24] Microsoft, “ASP.NET Overview,” p. 1, 2016. [Online]. Available: <https://msdn.microsoft.com/en-us/library/4w3ex9c2.aspx> [Accessed: 2017-07-17]
- [25] Flask, “Welcome — Flask (A Python Microframework),” 2014. [Online]. Available: <http://flask.pocoo.org/> [Accessed: 2017-07-17]
- [26] J. Deacon, “Model-View-Controller (MVC) Architecture,” *JOHN DEACON Computer Systems Development, Consulting & Training*, pp. 1–6, 2009. [Online]. Available: <http://www.jdl.co.uk/briefings/index.html{#}mvc>
- [27] D. S. Foundation, “Django Documentation,” pp. 1–1172, 2012. [Online]. Available: <https://docs.djangoproject.com/en/1.3/> [Accessed: 2017-07-17]
- [28] A. Holovaty and J. Kaplan-Moss, *The Definitive Guide to Django: Web Development Done Right*, 2009.
- [29] M. Banzi, D. Cuartielles, T. Igoe, G. Martion, and D. Mellis, “Arduino - Introduction,” 2012. [Online]. Available: <http://arduino.cc/en/Guide/Introduction> [Accessed: 2017-05-25]
- [30] “Arduino Nano - User Manual.” [Online]. Available: <https://www.arduino.cc/en/uploads/Main/ArduinoNanoManual23.pdf>
- [31] J. Melorose, R. Perroy, and S. Careas, “Arduino Nano,” 2015. [Online]. Available: <http://www.farnell.com/datasheets/1682238.pdf> [Accessed: 2017-07-17]
- [32] Raspberry Pi Foundation, “Raspberry Pi Foundation - About Us,” 2012. [Online]. Available: <https://www.raspberrypi.org/about/> [Accessed: 2017-07-17]

- [33] REOTEMP Instrument Corporation, “Thermocouple-Thermocouples-What is a thermocouple-Types of thermocouples.” [Online]. Available: <http://www.thermocoupleinfo.com/index.htm> [Accessed: 2017-07-17]
- [34] “Temperature Sensors Watlow Educational Series.” [Online]. Available: https://kontrolotomasyon.files.wordpress.com/2016/10/dt_{_}temperaturesensors_{_}thewatlow_{_}25ekim2016.pdf
- [35] Argus, “Light and lighting contril in greenhouses,” no. August, pp. 1–29, 2010. [Online]. Available: <http://www.arguscontrols.com/resources/Light-and-Lighting-Control-in-Greenhouses.pdf>
- [36] “Salinity Sensor (Order Code SAL-BTA) Collecting Data with the Salinity Sensor.” [Online]. Available: <https://www.vernier.com/files/manuals/sal-bta.pdf>
- [37] A. B. A. Rahman, “Comparison of Internet of Things (IoT) Data Link Protocols,” pp. 1–21, 2015. [Online]. Available: <http://www.cse.wustl.edu/{~}jain/cse570-15/index.html>
- [38] R. Bruno, M. Conti, and E. Gregori, “Bluetooth : Architecture , Protocols and Scheduling Algorithms,” *Cluster Computing*, vol. 5, no. 2, pp. 117–131, 2002. [Online]. Available: <http://link.springer.com/10.1023/A:1013989524865>
- [39] Bluetooth(TM), “Bluetooth Specification,” *Specification of the Bluetooth System*, vol. 1, p. 1084, 2001. [Online]. Available: <http://www.bluetooth.com>
- [40] W. Paper, “IEEE 802.11g The New Mainstream Wireless LAN Standard,” 2005. [Online]. Available: <http://www.broadcom.com/products/index>
- [41] “IEEE 802.11g — Wi-Fi WLAN — Tutorial - Radio-Electronics.Com.” [Online]. Available: <http://www.radio-electronics.com/info/wireless/wi-fi/ieee-802-11g.php> [Accessed: 2017-07-18]
- [42] “Sigfox Technology Overview — Sigfox.” [Online]. Available: <https://www.sigfox.com/en/sigfox-iot-technology-overview> [Accessed: 2017-07-18]
- [43] P. T. Hiep, H. Noi, V. Nam, N. H. Hoang, H. Noi, and V. Nam, “A Review of Open Source Software Development Life Cycle Models,” *International Journal of Multimedia and Ubiquitous Engineering*, vol. 9, no. 5, pp. 391–402, 2014. [Online]. Available: <http://dx.doi.org/10.14257/ijseia.2014.8.3.38>
- [44] Laudon, C. Kenneth, Laudon, and P. Jane, *Management Information Systems New Approaches to Organization & Technology*. Prentice Hall, 1998.

- [45] E. Turban, *Information technology for management : improving quality and productivity*. Wiley, 1996. [Online]. Available: https://books.google.pt/books?id=FqxzQgAACAAJ{%&}redir{%_}esc=y{%&}hl=pt-PT
- [46] “Gravatar - Globally Recognized Avatars.” [Online]. Available: <http://pt.gravatar.com/site/implementhttps://secure.gravatar.com/> [Accessed: 2017-06-10]
- [47] A. Rodriguez, “RESTful Web services: The basics,” no. February, pp. 1–11, 2015. [Online]. Available: <https://pdfs.semanticscholar.org/ca66/561d3602f65aef1301145e4e2689681b1967.pdfhttp://www.ibm.com/developerworks/webservices/library/ws-restful/>
- [48] T. Christie, “Django REST framework TokenAuthentication,” 2016. [Online]. Available: <http://www.django-rest-framework.org> [Accessed: 2017-07-03]
- [49] Ado Kukic, “Cookies vs Tokens: The Definitive Guide,” 2016. [Online]. Available: <https://auth0.com/blog/cookies-vs-tokens-definitive-guide> [Accessed: 2017-07-03]
- [50] SmartBear Software, “Swagger – The World’s Most Popular Framework for APIs.” 2017. [Online]. Available: <http://swagger.io/> [Accessed: 2017-06-07]
- [51] The Apache Software Foundation, “Foundation Project,” 2016. [Online]. Available: <https://www.apache.org/foundation/http://apache.org/foundation/> [Accessed: 2017-06-12]
- [52] “Datasheet, NTC Thermistor TTC05 Series, Disc Type for Temperature Sensing/Compensation.” [Online]. Available: <http://extra-parts.com/datasheets/TTC.pdf>
- [53] L. LIDA OPTICAL&ELECTRONIC CO., “Datasheet, CdS Photoconductive cells, GL5528,” p. 1. [Online]. Available: <https://pi.gate.ac.uk/pages/airpi-files/PD0001.pdf> [Accessed: 2017-05-24]
- [54] L. Guangzhou HC Information Technology Co ., “HC06 Datasheet,” no. 13, pp. 1–17, 2011.
- [55] Itseez, “About - OpenCV library.” [Online]. Available: <http://opencv.org/about.html> [Accessed: 2017-05-19]
- [56] H. A. Razavi and T. R. Kurfess, “Detection of Wheel and Workpiece Contact/Release in Reciprocating Surface Grinding,” *Journal of Manufacturing Science and Engineering*, vol. 125, no. 2, p. 394, 2003. [Online]. Available: <http://manufacturingscience.asmedigitalcollection.asme.org/article.aspx?articleid=1447160>
- [57] Nginx, “Welcome to NGINX Wiki! — NGINX,” 2017. [Online]. Available: <https://www.nginx.com/resources/wiki/> [Accessed: 2017-07-04]

-
- [58] N. Dalal and B. Triggs, “Histograms of Oriented Gradients for Human Detection,” *Proc. Int. Conf. Computer Vision and Pattern Recognition*, pp. 886–893, 2005. [Online]. Available: <http://lear.inrialpes.fr/people/triggs/pubs/Dalal-cvpr05.pdf>
- [59] “Feature Detection — OpenCV 2.4.13.2 documentation [ONLINE].” [Online]. Available: http://docs.opencv.org/2.4/modules/gpu/doc/object_{_}detection.html [Accessed: 2017-07-10]
- [60] A. Rosebrock, “HOG detectMultiScale parameters explained,” 2015. [Online]. Available: <http://www.pyimagesearch.com/2015/11/16/hog-detectmultiscale-parameters-explained/> [Accessed: 2017-07-11]
- [61] “Coarse Gaze Estimation.” [Online]. Available: http://www.robots.ox.ac.uk/ActiveVision/Research/Projects/2009bбенfold_{_}headpose/project.html{#}datasets [Accessed: 2017-07-11]

Interligação de componentes

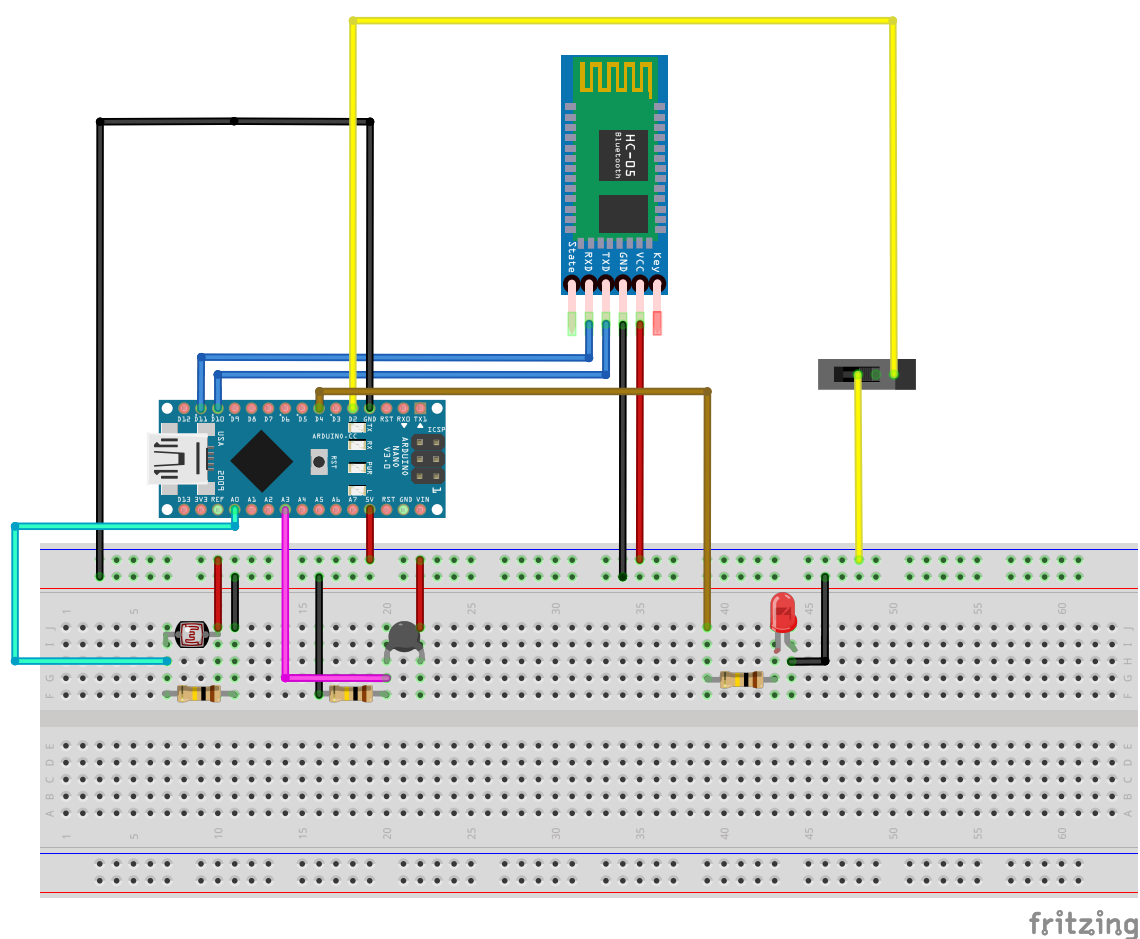


Figura A.1: Protótipo de montagem de componentes eletrotécnicos

