

ESCOLA SECUNDÁRIA RAUL PROENÇA

**Curso Profissional Técnico de Gestão e
Programação de Sistemas Informáticos**

RELATÓRIO DA PROVA DE APTIDÃO PROFISSIONAL (PAP)



Turma: 12.º CP-TGPSI

Aluno: Pedro Silva N.º 20

Professor Orientador: Luís Fernandes

Ano Letivo: 2021/2022

Índice

1. Introdução	6
1.1. Fundamentação da escolha do projeto	6
1.2. Finalidade do projeto (Objetivo)	6
1.3. Enquadramento do projeto	6
2. Desenvolvimento	8
2.1. Software utilizado	8
2.1.1. Plataforma Thingspeak	8
2.1.2. Arduino IDE	9
2.1.3. Visual Studio Code	10
2.2. Hardware utilizado	11
2.2.1. Módulo WiFi NodeMcuS NodeMcu	11
2.2.2. Sensor de pressão atmosférica e humidade – BME280	12
2.2.3. Sensor de LUX – BH1750	13
2.2.4. Sensor de temperatura – DS18B20	15
2.2.5. Sensor de direção do vento cata-vento	16
2.2.6. Sensor de chuva Pluviômetro	16
2.2.7. Sensor de velocidade do vento Anemômetro	17
2.2.8. Módulo Carregador de Baterias de Lítio - TP4056	17
2.2.9. Componentes diversos	19
2.3. Etapas de Desenvolvimento	21
2.3.1. Cronograma	21
2.3.2. Modelação 3D	22
2.3.3. Esquema geral de ligações	23
2.3.3. Descrição das fases do projeto	24
2.3.4. Estratégias adotadas na resolução de problemas	33
3. Conclusão	34
4. Referências	35

Índice de Figura

Imagem 1 - thingspeak	8
Imagem 2 - Canal do Thingspeak	9
Imagem 3 - Logo Arduino	10
Imagem 4 - Arduino IDE	10
Imagem 5 - Logo Visual Studio Code	11
Imagem 6 - Código do website	11
Imagem 7 - NodeMCU	13
Imagem 8 - Esquema de ligações do módulo wifi NodeMcu	13
Imagem 9 - BME280 sensor de humidade e pressão atmosférica	14
Imagem 10 - Esquema de ligações BME280	14
Imagem 11 - Esquema de ligações BH1750	16
Imagem 12 - Sensor de Temperatura DS18B20	17
Imagem 13 - Esquema de ligações DS18B20	17
Imagem 14 - Cata-vento	18
Imagem 15 - Pluviômetro	18
Imagem 16 - Anemômetro	18
Imagem 17 - Módulo de carregamento TP4056	20
Imagem 18 - Esquema de ligações do TP4056	20
Imagem 19 - Switch e condensador	21
Imagem 20 - Terminais de parafusos	21
Imagem 21 - Portas RJ11	21
Imagem 22 - Resistências	22
Imagem 23 - PCB	22
Imagem 24 - Cronograma	24
Imagem 25 - Bottom Plate	25
Imagem 26 - M6 Rods	25
Imagem 27 - Middle Rings	25
Imagem 28 - PCB Mount Frame	25
Imagem 29 - Screen Top Cover	25
Imagem 30 - Solar Panel Mount	25
Imagem 31 - Esquema geral de ligações	26
Imagem 32 - Impressão das peças 3D	27
Imagem 33 - Teste do sensor de temperatura DS18B20	28
Imagem 34 - Soldadura	28
Imagem 35 - PCB completa	29
Imagem 36 - Montagem da estrutura	29
Imagem 37 - Montagem da estrutura	30
Imagem 38 - Instalação das bibliotecas	30
Imagem 39 - Instalação das bibliotecas	30
Imagem 40 - Preferências Arduino IDE	31

Imagem 41 - Gestor de placas	31
Imagem 42 - Mudança para unidades do sistema Métrico	32
Imagem 43 - Primeiros resultados dos sensores	32
Imagem 44 - Código Arduino para o ThingSpeak	33
Imagem 45 - Criação do novo canal do thingspeak	34
Imagem 46 - Canal do Thingspeak	34
Imagem 47 - Código do Thingspeak para embeber as tabelas no website	35
Imagem 48 - Código do website	35
Imagem 49 - website	35
Imagem 50 - imagem do "screen top cover"	36

Agradecimentos

Agradeço aos meus professores pela ajuda, mas especialmente ao professor Luís Fernandes que sempre se disponibilizou na ajuda ao longo do ano, mas também a professora Carla Jesus que ajudou no relatório e a apresentação.

1. Introdução

O projeto escolhido para a realização da Prova de Aptidão Profissional (PAP), consiste na montagem de uma estação meteorológica e de um *website*, que permite visualizar os dados lidos por diversos sensores (humidade, temperatura, pressão atmosférica e outros).

A motivação para a realização desta estação meteorológica, está relacionada com o gosto pelo hardware em geral e, em particular da eletrónica e dos microcontroladores, assim como da programação deste tipo de dispositivos eletrónicos.

1.1. Fundamentação da escolha do projeto

A estação meteorológica proposta, teve por base o projeto da *instructables*: “Solar Powered WiFi Weather Station V3.0”. Após a análise deste projeto, decidiu-se implementar uma solução similar, mas adaptada às necessidades requeridas. Como por exemplo, a aplicação de algumas das temáticas abordadas em contexto sala de aula. Concretamente, o desenvolvimento baseado em microcontroladores, a soldadura em PCB (*Printed Circuit Board*) de componentes eletrónicos, a programação de *websites*, assim como, da organização e consequente implementação de um projeto como um nível de dificuldade superior. Ainda no âmbito do desenvolvimento deste projeto, foi elaborado um processo de investigação, nomeadamente, na utilização de novas ferramentas, como o IDE do Arduino e o Thingspeak.

1.2. Finalidade do projeto (Objetivo)

O objetivo deste projeto, consiste em disponibilizar aos utilizadores em geral, o acesso em tempo real, aos dados meteorológicos da Escola Secundária Raul Proença. Por intermédio de um website, que disponibiliza um conjunto gráficos e visualizações, como por exemplo, a temperatura, a humidade, a velocidade do vento e outros. Deste modo, propõe-se mitigar a ausência de dados meteorológicos da escola, beneficiando, deste feita, toda a comunidade educativa, com informação muito relevante no que concerne à escolha da sua roupa e de utensílios de proteção contra intempéries.

1.3. Enquadramento do projeto

Com este projeto, toda a comunidade educativa beneficia, tendo acesso a informação meteorológica local em tempo real, permitindo a tomada decisão da roupa a utilizar para uma visita à Escola Secundária Raul Proença.

As disciplinas que tiveram mais impacto na realização deste projeto, foram a arquitetura de computadores e a programação. Contudo, destaca-se a disciplina de arquitetura de computadores, pelo facto de nesta disciplina ter aprendido a programar para o Arduino e a soldar componentes eletrónicos num PCB, o que permitiu adquirir o conhecimento necessário para a construção deste projeto.

O relatório está dividido em 3 capítulos. No primeiro capítulo é abordado a descrição do hardware e software utilizados, no segundo capítulo o cronograma, esquema geral de ligações, descrição das fases do projeto e estratégias adotadas na resolução das

problemáticas encontradas. Finalmente, no terceiro capítulo, a conclusão, as referências bibliográficas e os anexos.

2. Desenvolvimento

2.1. Software utilizado

2.1.1. Plataforma Thingspeak

Thingspeak é uma plataforma *opensource* de análise IoT (Internet of Things) que permite agregar, visualizar e analisar streams de dados, de uma forma muito simples. Possibilita a recolha de dados em tempo real, a sua visualização de dados sobre a forma de gráficos, a criação de plugins e aplicações para colaboração com serviços web, redes sociais e outras interfaces de aplicações (API's) e a análise dos dados através da ferramenta Matlab.

Para a concretização do projeto, foi utilizado para a recolha de dados enviados pela estação meteorológica e respetiva visualização, por parte dos utilizadores, através da integração dos respetivos gráficos no webwebsite desenvolvido para o efeito.



Imagem 1 -thingspeak

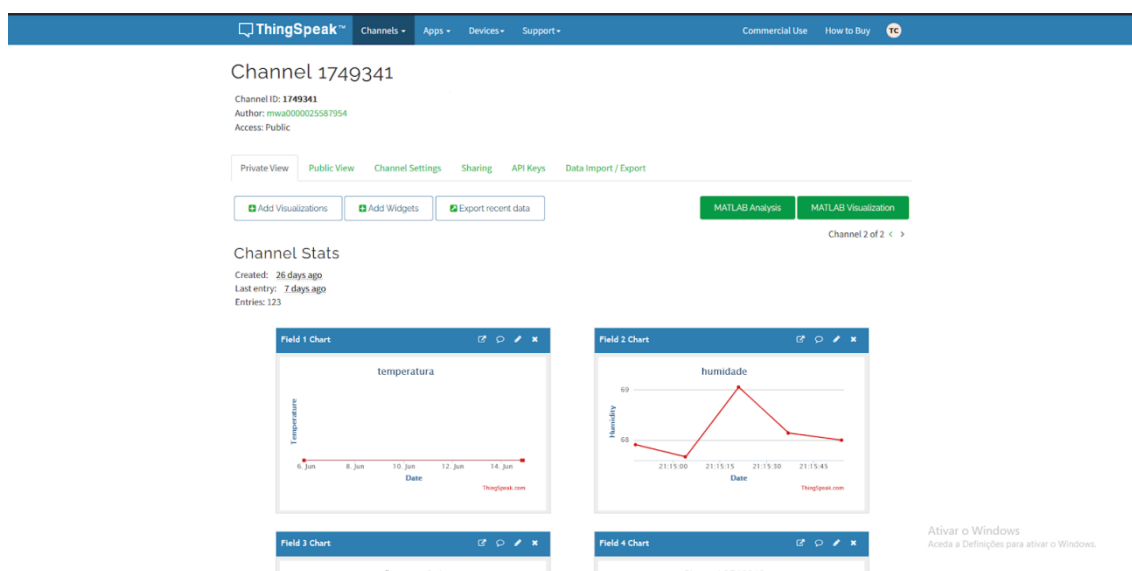


Imagem 2 - Canal do Thingspeak

2.1.2. Arduino IDE

O IDE (*Integrated Development Environment*) de desenvolvimento do Arduino, corresponde a uma plataforma *open source*, que permite escrever, editar, testar (*debug*) e compilar o código utilizado na programação de microcontroladores baseados no Arduino. Após a compilação do código fonte para código máquina, sem erros de compilação, o IDE permite enviar o binário para o microcontrolador em questão (placas baseadas no Arduino). Uma das grandes vantagens deste IDE, corresponde à utilização de bibliotecas, tipicamente, implementadas na linguagem de programação C/C++, que permitem a adição de funcionalidades extras, como por exemplo, a capacidade para controlar uma matriz de LED. Contudo, este IDE é compatível com vários tipos de placas de desenvolvimento, como por exemplo, a utilizada no meu projeto NodeMCU.



Imagem 3 - Logo Arduino



Imagem 4 - Arduino IDE

2.1.3. Visual Studio Code

O Visual Studio Code é um editor de código da Microsoft, que permite desenvolver em diversas linguagens de programação, como por exemplo, Python, Java, C++ e outras. Este editor, permite ainda o *debug* de código, controlo de versões (Git incorporado), realce de sintaxe, etc. Apresenta outra vantagem, que corresponde à instalação de extensões que facilitam a implementação de código.

Neste projeto, o Visual Studio Code foi utilizado para desenvolver um *website*, que permite apresentar gráficos e visualizações, que espelham os valores lidos pelos vários sensores utilizados nesta estação meteorológica.



Imagem 5 - Logo Visual Studio Code

```
<html lang="en">
<head>
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no" />
  <meta name="description" content="" />
  <meta name="author" content="" />
  <title>Scrolling Nav - Start Bootstrap Template</title>
  <link rel="icon" type="image/x-icon" href="assets/favicon.ico" />
  <!-- Core theme CSS (includes Bootstrap) -->
  <link href="css/styles.css" rel="stylesheet" />
</head>
<body id="page-top">
  <!-- Navigation -->
  <nav class="navbar navbar-expand-lg navbar-dark bg-dark fixed-top" id="mainNav">
    <div class="container px-4">
      <a class="navbar-brand" href="#page-top">Meteriologia Raul</a>
      <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarResponsive" aria-controls="navbar"
      <div class="collapse navbar-collapse" id="navbarResponsive">
        <ul class="navbar-nav ms-auto">
          <li class="nav-item"><a class="nav-link" href="#about">Sobre</a></li>
          <li class="nav-item"><a class="nav-link" href="#services">Dados meteriologicos</a></li>
```

Imagem 6 - Código do website

2.2. Hardware utilizado

2.2.1. Módulo WiFi NodeMcu

O NodeMCU tem um regulador de 3,3V (corrente max 1A) , esta especificado que a corrente fornecida para alimentação de 3,3V deve ser no mínimo de 500 mA.

A alimentação desta placa pode ser feita através do próprio conector USB (5,0V) ou então através do pino 5V ou VIN , com uma alimentação regulada de 5,0V.

Ele suporta USB 2.0 com velocidade de até 1 MBps. Na placa NodeMcu, a alimentação 3,3V do CP2102 é feita através do regulador AMS1117. Consumo máximo de corrente é de apenas 26 mA. O CP2102 possui um regulador interno de 3,3V, mas nesta placa este regulador esta inoperante.

Características:

- CPU: Xtensa® Dual-Core 32 bits LX6;
- Processador: Xtensa 32-Bit LX6 Dual Core;
- ROM: 448 KBytes;
- RAM: 520 KBytes;
- Flash: 4Mb;
- Relógio Máximo: 240MHz;
- Padrão sem fio 802.11 b/g/n;
- Conexão WiFi 2.4GHz (Máximo de 150 Mbps);
- Antena Embutida;
- Conector MicroUSB;
- WiFi Direct (P2P), P2P Discovery, P2P Group Owner Mode e P2P Power Management;
- Modos de Operação: STA/AP/STA+AP;
- Bluetooth BLE 4.2;
- Portas GPIO: 11;
- Taxa de Transferência: 110-460800bps;
- Suporta Upgrade Remoto de Firmware;
- Conversor Analógico Digital (ADC);
- Distância Entre Pinos: 2,54mm;
- Dimensões: 49x26x7mm.

O NodeMCU é usado neste projeto para recolher os dados dos sensores e para transmiti-los para o *thingspeak*.

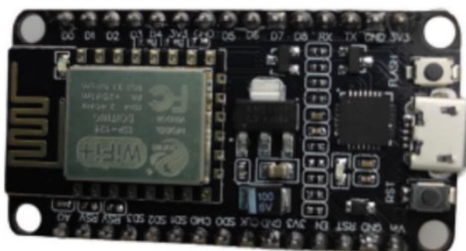


Imagem 7 - NodeMCU

Esquema de ligações do modulo wifi NodeMcu

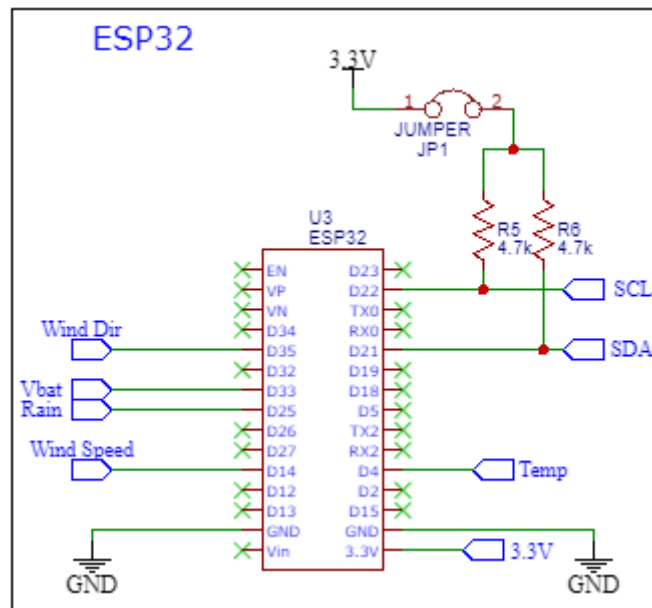


Imagem 8 - Esquema de ligações do modulo wifi NodeMcu

2.2.2. Sensor de pressão atmosférica e humidade – BME280

O sensor atmosférico BME280 mede a pressão atmosférica de 30kPa até 110kPa, bem como a humidade e a temperatura. Este fornece uma interface SPI de 3.3V, uma interface I2C tolerante a 5V (com resistencia pull-up para 3.3V).

Este sensor de precisão é a melhor solução de deteção de baixo custo para medir humidade com precisão de $\pm 3\%$, pressão atmosférica com precisão absoluta de ± 1 hPa e temperatura com precisão de $\pm 1,0^\circ\text{C}$.

Características:

- Tensão de operação: 5VDC (Possui regulador 3,3V on-Board)
- Nível de sinais de saída: 3,3V
- Faixa de humidade: 0 a 100%;
- Faixa de Temperatura: -40 a 85°C ;
- Faixa de Pressão: 300 a 1100hPa;
- Resolução da humidade: 0,008%;
- Resolução da temperatura: $0,01^\circ\text{C}$;
- Precisão da temperatura: $\pm 1^\circ\text{C}$ ($0\sim 65^\circ\text{C}$)
- Precisão da pressão: ± 1 Pa;
- Interface: I2C e SPI
- Comunicação: I2C (até 3,4MHz)
- Dimensões (CxL): 13,5x10mm;

Sensor de pressão atmosférica e humidade

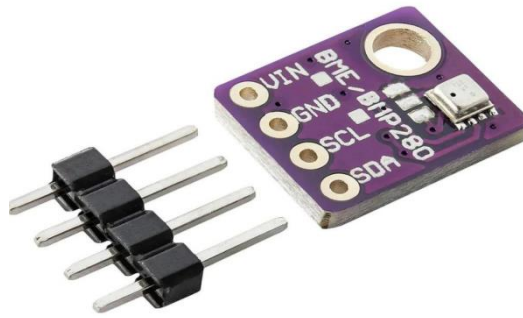


Imagem 9 - BME280 sensor de humidade e pressão atmosférica

Esquema de ligações do sensor de temperatura BME 280

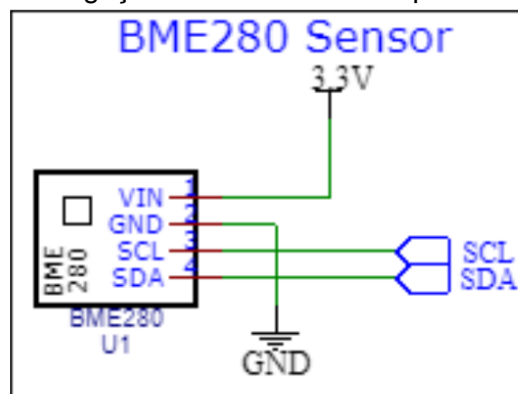


Imagem 10 - Esquema de ligações BME280

2.2.3. Sensor de LUX – BH1750

O BH1750 é um sensor de luz ambiente digital. Este sensor pode medir com precisão o valor LUX da luz até 65535lx. Este sensor usa um fotodiodo para detetar a luz. Devido ao efeito fotoelétrico interno, eletricidade é produzida no fotodiodo. Esta eletricidade produzida é proporcional à intensidade da luz.

LUX - unidade sistema internacional de iluminação, igual a um lúmen por metro quadrado

Características:

- Espectro do Sensor IR: Comprimento de onda: 550nm-1000nm (centrado em 800)
- Espectro do Sensor de Luz Visível: Comprimento de onda: 400nm-800nm (centrado em 530)
- Fonte de Tensão: Alimentação com DC 3.3-5V
- Tipo de saída: endereço I2C 0x60 (7 bits)
- Temperatura de operação: -40°C ~ 85°C
- Tamanho da placa: 13,4x10,6mm
- Peso:0,8g

Sensor de LUX BH1750

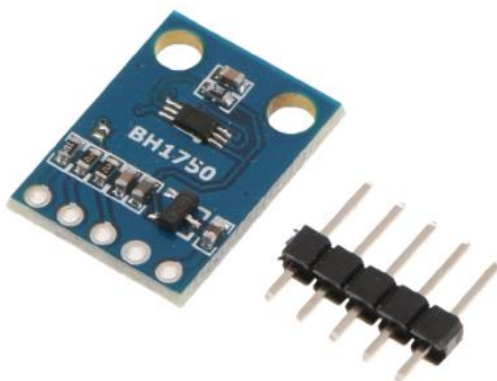


Imagem 11- Sensor de LUX BH1750

Esquema de ligações do sensor de LUX BH1750

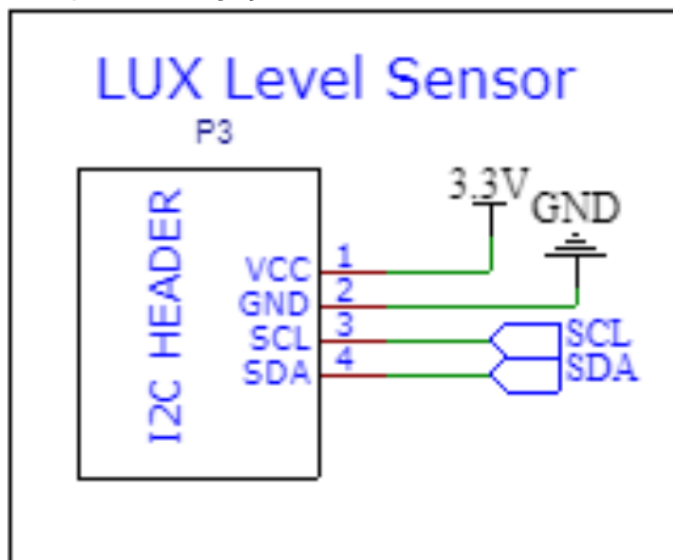


Imagem 11 - Esquema de ligações BH1750

2.2.4. Sensor de temperatura – DS18B20

O Sensor de Temperatura DS18B20 é um componente eletrónico digital desenvolvido para ser aplicado nos mais diversos ambientes, pois é capaz de medir a temperatura em locais húmidos, incluindo de baixo de água.

O Sensor de Temperatura DS18B20 tem a precisão de $\pm 0.5^{\circ}\text{C}$, e proporciona leituras de temperatura de até 12-bits através de uma conexão de dados de apenas 1 fio.

Características:

- Chip: DS18B20;
- Tensão da operação: 3-5,5V;
- Faixa de medição: -55°C a $+125^{\circ}\text{C}$;
- Precisão: $\pm 0.5^{\circ}\text{C}$ entre -10°C e $+85^{\circ}\text{C}$;
- Ponta de aço inoxidável;
- Dimensão ponta de aço: 6 x 50mm;
- Dimensão do cabo: 100cm;
- Interface de 1 fio.
- Dimensão ponta de aço: 6 x 50mm
- Dimensão do cabo: 100cm
- Interface de 1 fio

Sensor de Temperatura DS18B20



Imagem 12 - Sensor de Temperatura DS18B20

Esquema de ligações DS18B20.

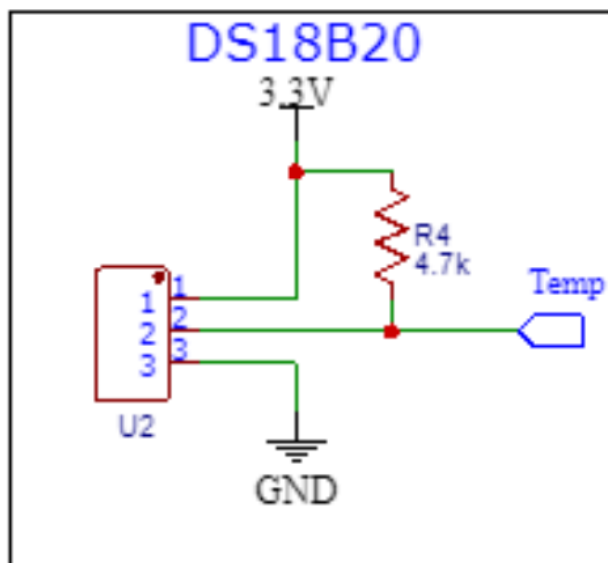


Imagem 13 - Esquema de ligações DS18B20

2.2.5. Sensor de direção do vento cata-vento



Imagem 14 - Cata-vento

2.2.6. Sensor de chuva Pluviômetro



Imagem 15 - Pluviômetro

2.2.7. Sensor de velocidade do vento Anemômetro



Imagem 16 - Anemômetro

2.2.8. Módulo Carregador de Baterias de Lítio - TP4056

O módulo TP4056 consiste num carregador linear de corrente e tensão constantes para baterias de lítio recarregáveis, de uma só célula. Além de carregar com segurança uma bateria de lítio, o módulo fornece, também, a proteção necessária exigida pelas baterias de lítio, nomeadamente:

- Proteção contra descarga excessiva - evita que a bateria seja descarregada abaixo de 2,4 V, o nível de tensão mínimo saudável para a bateria;
- Proteção contra sobrecarga - o módulo carregará a bateria com segurança para 4,2V
- Proteção contra sobrecorrente e curto-circuito - o módulo cortará a saída da bateria se a taxa de descarga exceder os 3A ou se ocorrer um curto-circuito.
- Carga lenta (recondicionamento da bateria) - se o nível de tensão da bateria conectada for inferior a 2,9V, o módulo utilizará uma corrente de carga lenta de 130mA até que a tensão da bateria atinja os 2,9V, ponto em que a corrente de carga será aumentada linearmente para a corrente de carga configurada.

O fornecimento de energia, para carregamento da bateria, pode ser efetuado a partir de um cabo micro USB ou através das conexões + e -. A fonte de alimentação terá de ser capaz de fornecer pelo menos 1A para que o carregador carregue corretamente a bateria.

Possui dois LEDs indicadores, o vermelho indica que a bateria está em carregamento e o azul indica que a carga da bateria está completa.

Este módulo utiliza uma quantidade muito pequena de corrente, na ordem dos microamperes, sempre que é conectado a uma bateria. Pode ser conectado a uma bateria por longos períodos de tempo, embora não seja aconselhável deixar o módulo ligado à bateria, sem carregamento, por períodos superiores a quatro meses.

Características:

- Controlador de carga: TP4056
- Proteção IC: DW01A
- MOSFET de controle de carga/descarga: FS8205A
- Método de Carga: Corrente Constante/Tensão Constante (CC/CV)
- Tensão de alimentação de entrada: 4,5~6,0 V
- Corrente de carga constante: 1 A
- Tensão de carga completa (flutuante): $4,2\text{ V} \pm 1,5\%$
- Proteção contra sobrecarga, descarga excessiva e sobrecorrente
- Carga lenta (recondicionamento da bateria)
- Dimensões: 28,5 x 17,3 x 3,6mm
- Peso: 1,6g

Módulo de carregamento TP4056

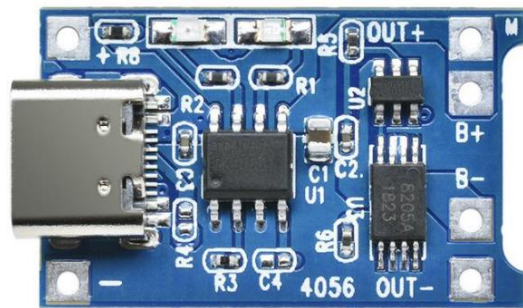


Imagem 17 - Módulo de carregamento TP4056

Esquema de ligações do TP4056

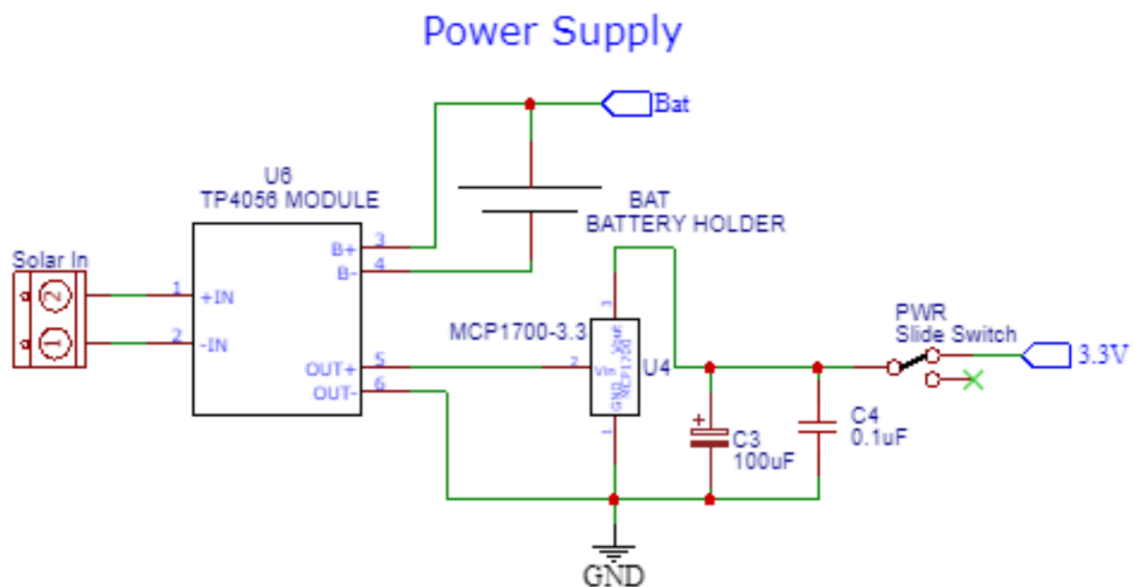


Imagem 18 - Esquema de ligações do TP4056

2.2.9. Componentes diversos

À esquerda o *switch* que serve para desligar e ligar a estação quando o interruptor é desligado, o circuito é interrompido e o fluxo de energia é interrompido. E à direita o condensador eletrolítico que armazena e libera eletricidade no circuito, à medida que a corrente contínua flui para um condensador, ele carrega energia e libera um fluxo de corrente alternada de volta ao circuito que serve para evitar picos de voltagem.

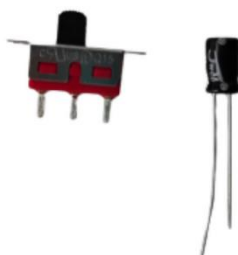


Imagem 19 - Switch e condensador

Os Terminais de parafusos servem para ligar o painel solar e o sensor de temperatura DS18B20 ao NodeMCUe ao TP4056.



Imagem 20 - Terminais de parafusos

Conector RJ11, que vão servir para ligar o cata-vento, o Anemómetro e o Pluviómetro à PCB

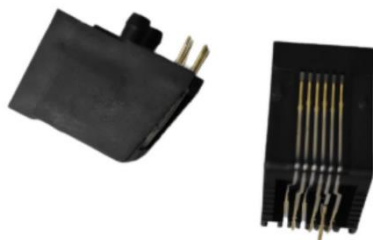


Imagem 21- Portas RJ11

Uma resistência é um dispositivo elétrico, com objetivo de transformar energia elétrica em energia térmica por meio do efeito joule, e serve para limitar a corrente elétrica em um circuito. Também pode ser usado para regular a corrente elétrica para fornecer a corrente específica para um componente.

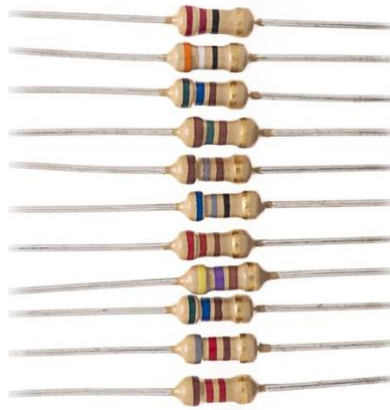


Imagem 22- Resistências

PCB

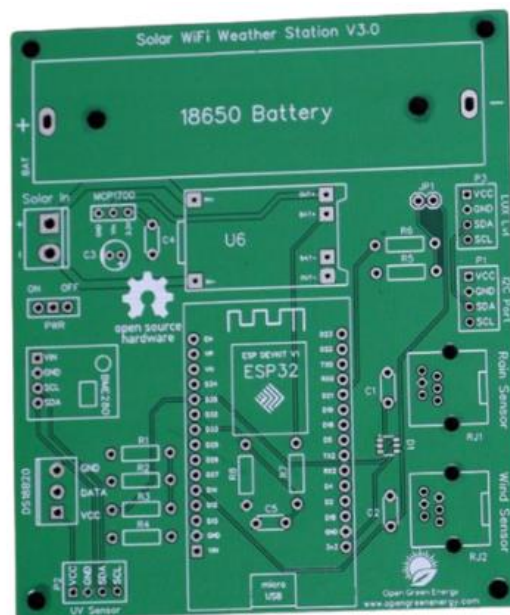


Imagem 23 - PCB

2.3. Etapas de Desenvolvimento







2.3.1. Cronograma

O cronograma mostra as atividades realizadas e quando foram realizadas

Itens	Meses		Novembro				Dezembro				Janeiro				Fevereiro				Março				Abril				Maio				Junho						
	semana		s1	s2	s3	s4	s5	s6	s7	s8	s9	s10	s11	s12	s13	s14	s15	s16	s17	s18	s19	s20	s21	s22	s23	s24	s25	s26	s27	s28	s29	s30	s31	s32	s33	s34	s35
Pesquisa																																					
Planificação																																					
Inventário de materiais e esperar que eles cheguem																																					
Relação dos materiais																																					
Montagem de equipamento																																					
Desenvolvimento do código																																					
Testes																																					
Escrita do Relatório																																					
Desenvolvimento da Apresentação																																					

Imagem 24 - Cronograma

2.3.2. Modelação 3D

 <p>Imagem 25- Bottom Plate</p>	 <p>Imagem 26 - M6 Rods</p>	 <p>Imagem 27 - Middle Rings</p>
 <p>Imagem 28 - PCB Mount Frame</p>	 <p>Imagem 29 - Screen Top Cover</p>	 <p>Imagem 30 - Solar Panel Mount</p>

2.3.3. Esquema geral de ligações

Esquema geral de ligações

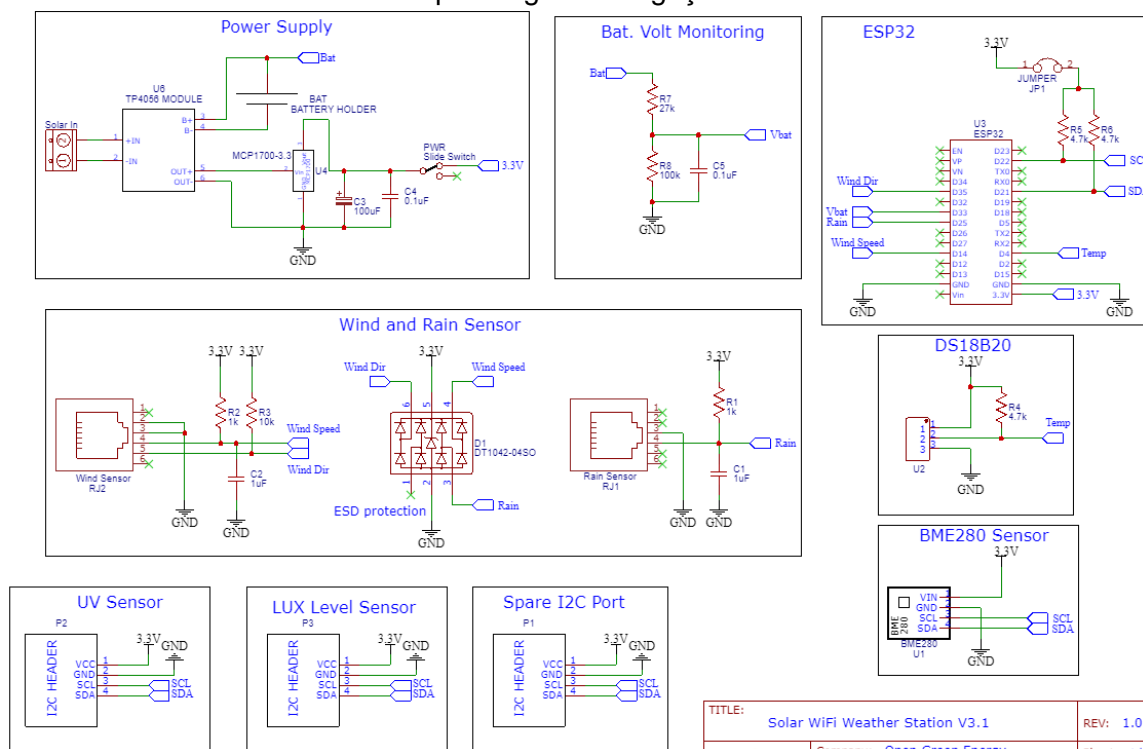


Imagem 31 - Esquema geral de ligações

2.3.4. Descrição das fases do projeto

Pesquisa de possíveis projetos

Foi realizada uma pesquisa e consequente análise de diversos projetos no *website instructables*, com o objetivo de recolher um conjunto de propostas para a elaboração da PAP. Após a apresentação de alguns projetos, foi escolhido o: “*Solar Powered WiFi Weather Station V3.0*”.

Aquisição dos componentes necessários

O material foi encomendado no mês de dezembro em varias lojas como PTRobotics, castro eletrónica e a plataforma Aliexpress mas houve um atraso com alguns componentes, que só chegaram no inicio de abril.

Impressão 3D

Este projeto foi baseado nos modelos feitos pelo criador do projeto que estão disponibilizados no *thingiverse*, utilizando a impressora *Anet A6* com o software *simplify 3d* e o filamento PLA de 1.75mm branco, no total demorou 1 semana para imprimir .

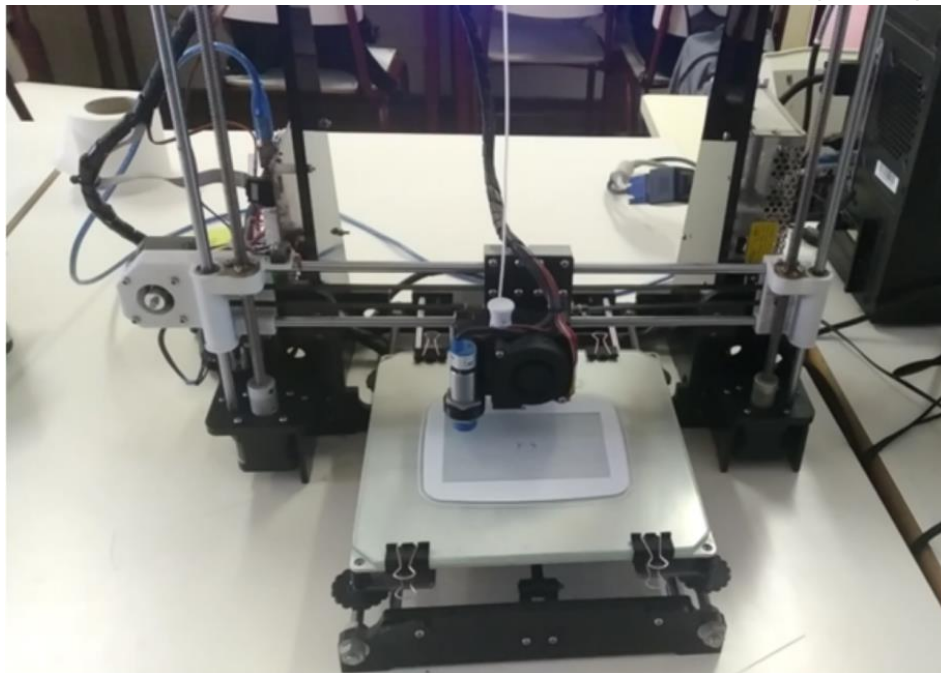


Imagem 32 - Impressão das peças 3D

Testes

Antes da montagem foram testados os sensores de temperatura (DB18B20) e de humidade e pressão atmosférica (BME280), com recurso à placa de desenvolvimento Arduino UNO e um *breadboard*.

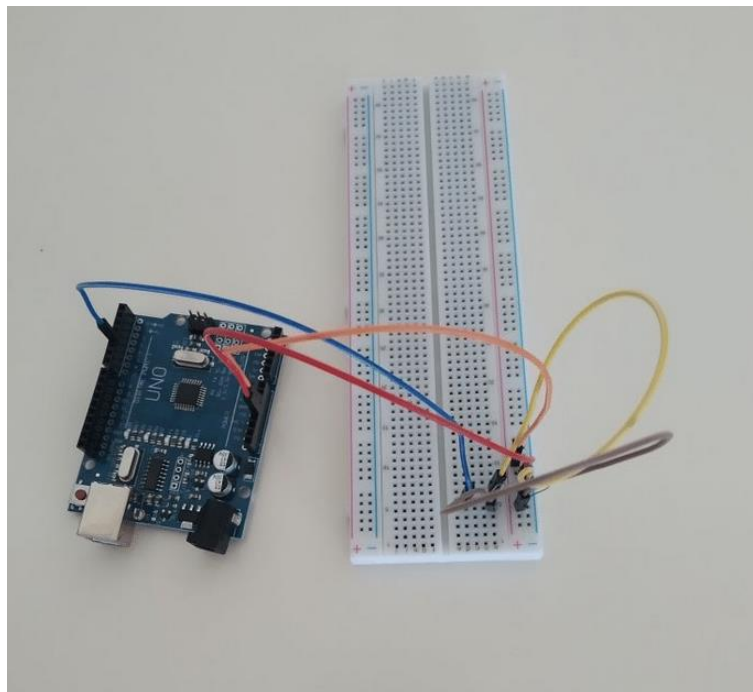


Imagem 33 - Teste do sensor de temperatura DB18B20

Soldadura

Depois da impressão 3D e da maior parte dos competentes eletrónicos terem chegado, deu-se início à soldadura na placa de circuito impresso (PCB). De modo, a facilitar este processo, em primeiro lugar, foram soldados os componentes eletrónico mais pequenos.

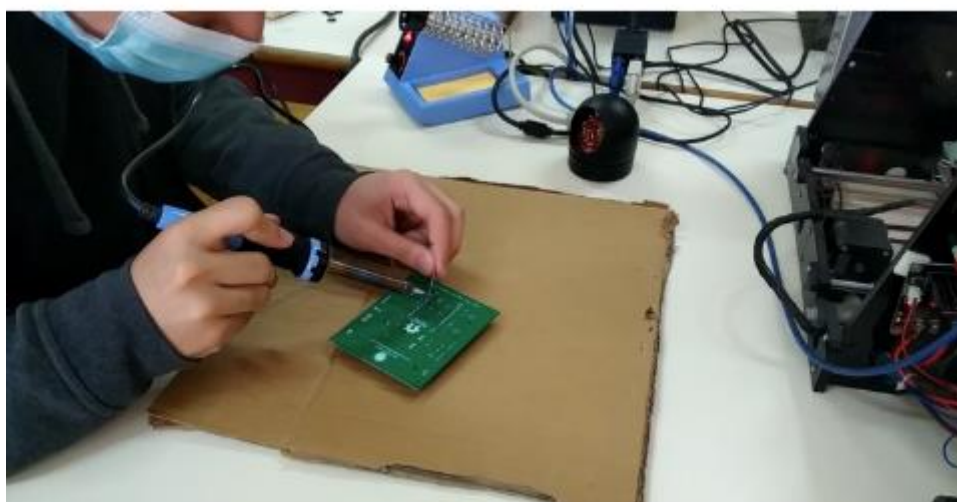


Imagem 34 - Soldadura

PCB com todos os componentes soldados



Imagem 35 - PCB completa

Montagem

Montagem dos “Middle rings” nos “M6 rods”



Imagem 36 - Montagem da estrutura

Montagem da “Screen top cover” nos “M6 RODS”



Imagem 37 - Montagem da estrutura

Programação Arduino

O primeiro passo da programação foi a instalação das bibliotecas necessárias: onewireNG, Adafruit BME280, BH1750, Dallas Temperature e Adafruit SI1145, e a instalação da placa ESP32

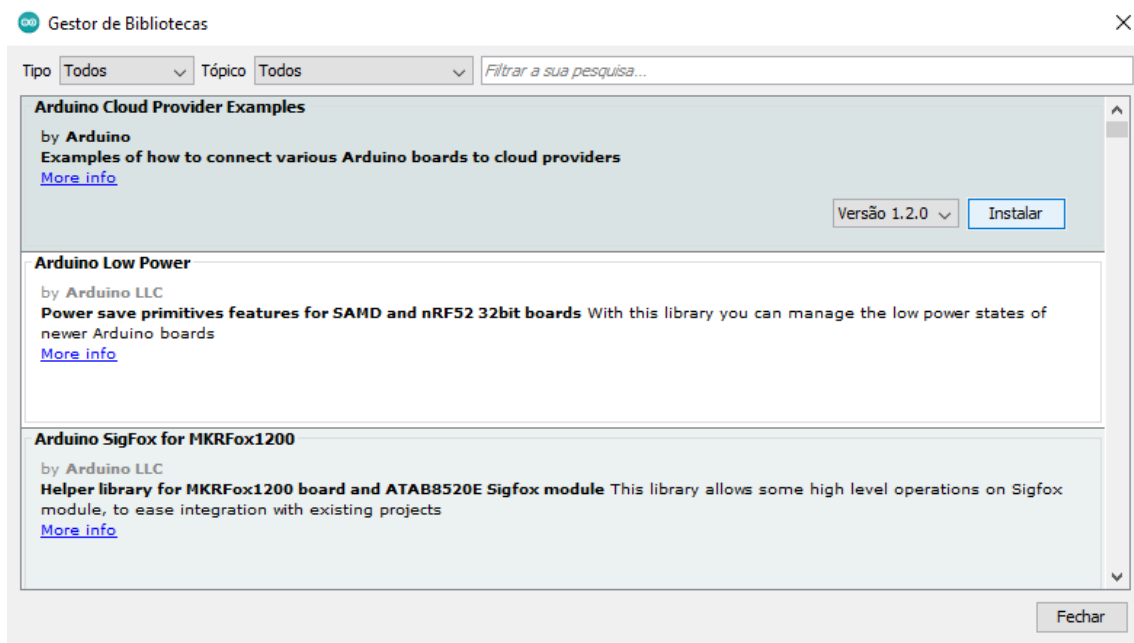


Imagem 38 - Instalação das bibliotecas

Imagem 39 - Instalação das bibliotecas

Para instalar a placa ESP32 foi necessário copiar o link do repositório para a pagina das preferências e depois instalar no gestor de placas .

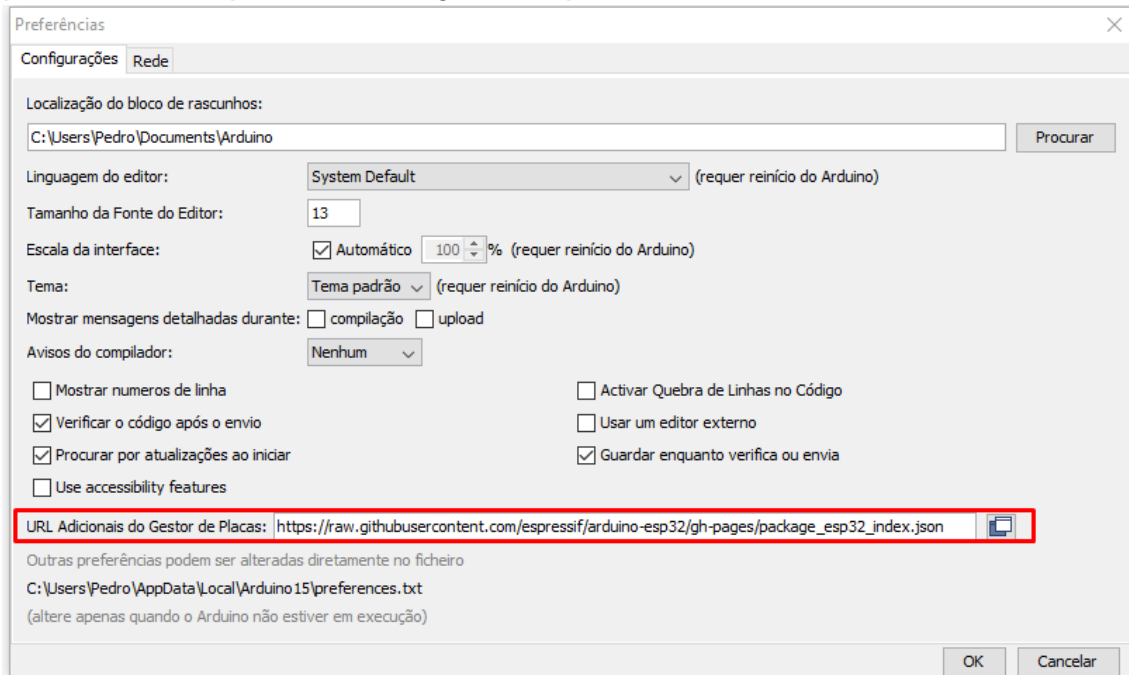


Imagem 40 - Preferências Arduino IDE



Imagem 41 - Gestor de placas

Após a instalação das bibliotecas e das placas foi preciso mudar as linhas de código de escolha de sistema de medição dos sensores.

```
//=====
//Metric or Imperial measurements
//=====
#define METRIC
```

Imagem 42 - Mudança para unidades do sistema Métrico

Também foi preciso reinstalar algumas bibliotecas com versões mais antigas devido a algumas incompatibilidades e resolver alguns erros no código.

Depois das alterações foi só preciso enviar para o NodeMCUe testar a ligação ao thingspeak.

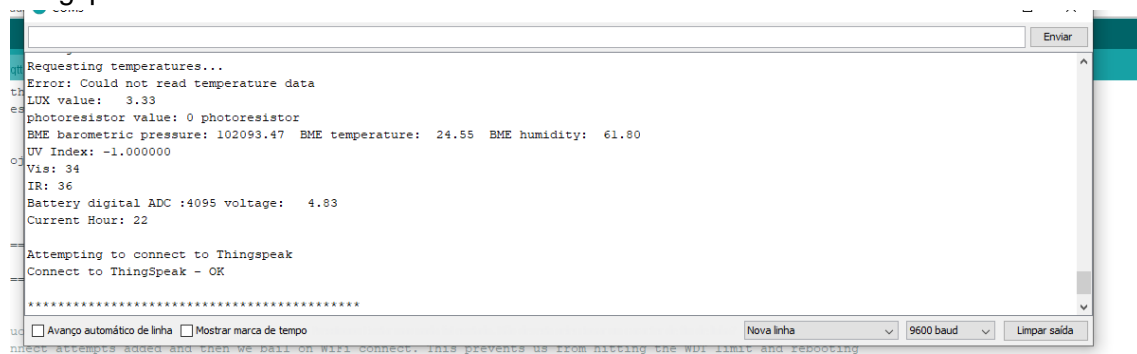


Imagem 43 - Primeiros resultados dos sensores

Thingspeak

Código do Arduino referente ao Thingspeak

```
else if (App == "Thingspeak")
{
    Serial.println("Attempting to connect to Thingspeak");
    // Send data to ThingSpeak
    WiFiClient client;
    if (client.connect(thingspeak_server, 80))
    {
        Serial.println("Connect to ThingSpeak - OK");
        Serial.println("");
        Serial.println("*****");

        int hourPtr = timeinfo.tm_hour;

        String postStr = "";
        postStr += "GET /update?api_key=";
        postStr += api_key;
        postStr += "&field1=";
        postStr += String(environment->batteryVoltage);
        postStr += "&field2=";
        postStr += String(environment->humidity);
        postStr += "&field3=";
        postStr += String(environment->barometricPressure);

        postStr += "&field4=";
#ifdef METRIC
        postStr += String(environment->temperatureC);
#else
        postStr += String(environment->temperatureF);
#endif
    }
}
```

Imagem 44 - Código Arduino para o ThingSpeak

Depois da programação do Arduino, deu-se início à aprendizagem da plataforma thingspeak. Em primeiro lugar, foi criada uma conta da plataforma thingspeak, em seguida, foi criado um canal, com oito campos, para a receção dos dados lidos nos sensores (temperatura, humidade, pressão atmosférica, diferença de voltagem da bateria, LUX, direção do vento, velocidade do vento e precipitação). Por último, foram definidos os gráficos e as visualizações.

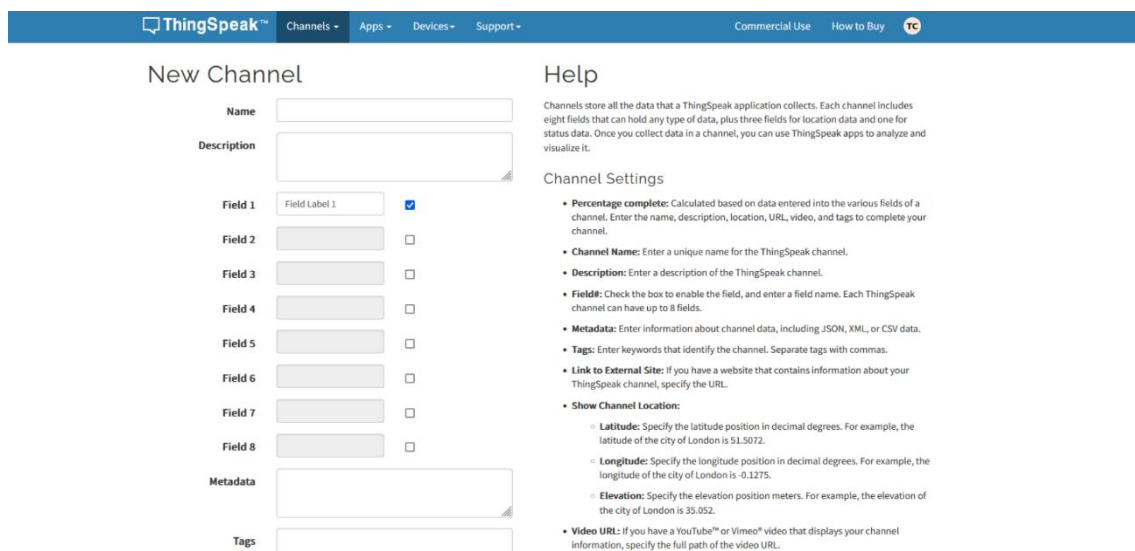


Imagem 45 - Criação do novo canal do thingspeak

Canal do Thingspeak

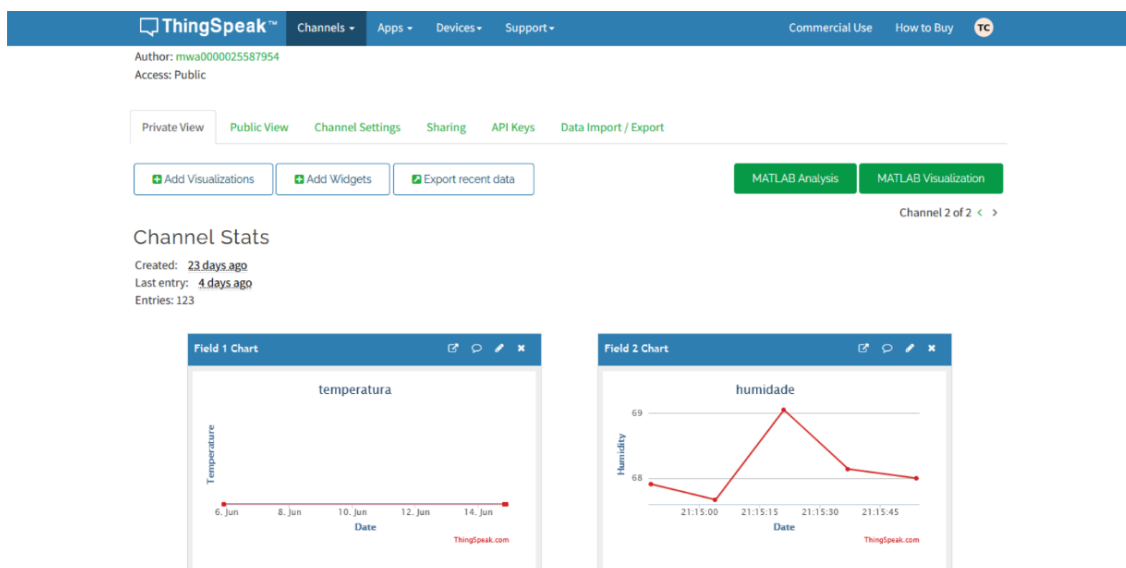


Imagem 46 - Canal do Thingspeak

Website

Código do Thingspeak para embeber as tabelas no website

Field 3 Chart IFrame

```
<iframe width="450" height="260" style="border: 1px solid #cccccc;" src="https
```

Imagem 47 - Código do Thingspeak para embeber as tabelas no website

Código das tabelas do Thingspeak embebido no visual studio code

```
<section class="bg-light" id="services">

  <div class="center">
    <iframe width="450" height="260" style="border: 1px solid #cccccc;" src="https://thingspeak.com/channels/1749341/charts/1?bgcolor=%23ffffff&color=%23d62728" />
    <iframe width="450" height="260" style="border: 1px solid #cccccc;" src="https://thingspeak.com/channels/1749341/widgets/486406" />
  </div>

  <div class="center">
    <iframe width="450" height="260" style="border: 1px solid #cccccc;" src="https://thingspeak.com/channels/1749341/charts/2?bgcolor=%23ffffff&color=%23d62728" />
    <iframe width="450" height="260" style="border: 1px solid #cccccc;" src="https://thingspeak.com/channels/1749341/widgets/486409" />
  </div>

  <div class="center">
    <iframe width="450" height="260" style="border: 1px solid #cccccc;" src="https://thingspeak.com/channels/1749341/charts/2?bgcolor=%23ffffff&color=%23d62728" />
    <iframe width="450" height="260" style="border: 1px solid #cccccc;" src="https://thingspeak.com/channels/1749341/widgets/485968" />
  </div>

</section>
```

Imagem 48 - Código do website

Resultado final do website

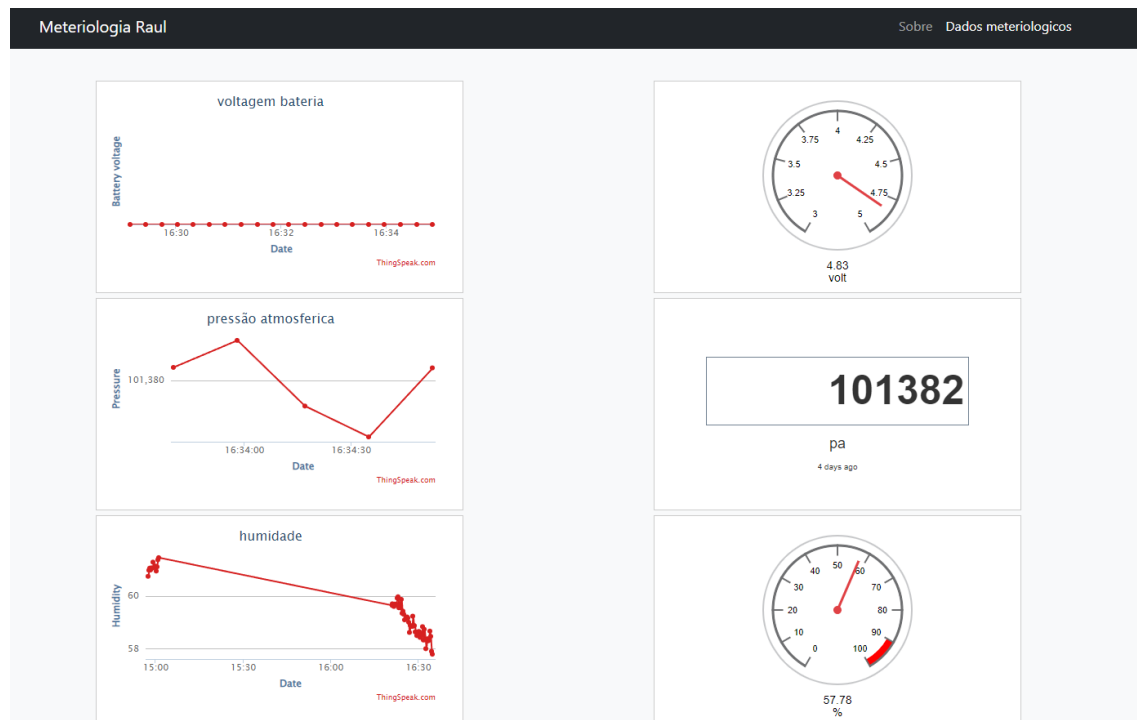


Imagem 49 - website

2.3.5. Estratégias adotadas na resolução de problemas

O maior problema no projeto foi os “m6 roods” não encaixarem nos buracos dos “middle rings” por serem demasiado grandes, a solução foi limar os “m6 roods” até que encaixassem nos buracos.



Imagem 50 - imagem do "screen top cover"

Também foi preciso reinstalar algumas bibliotecas com versões antigas devido a algumas incompatibilidades com outras bibliotecas. E também foi preciso mudar a biblioteca “OneWire” para a biblioteca “OneWireNG”.

3. Conclusão

Em resumo, o desenvolvimento deste projeto contribuiu para as minhas competências técnicas, nomeadamente, a programação de microcontroladores, a gestão de tempo e a soldadura de componentes eletrónicos.

Consideram-se, que os objetivos inicialmente propostos, foram atingidos. Contudo, no desenvolvimento deste projeto, surgiram alguns problemas, que foram ultrapassados com o apoio e ajuda dos professores. Como por exemplo, a entrega de alguns componentes eletrónicos (portas RJ11).

Pretende-se continuar a trabalhar nesta estação meteorológica, pelo facto, de a mesma vir a ser implementada na escola. Considerando-se uma mais valia, para toda a comunidade educativa.

De modo geral, gostei muito de desenvolver este projeto. Contudo, senti alguma dificuldade na elaboração deste relatório final, porque no decurso do desenvolvimento desta estação meteorológica, surgiram vários problemas com alguns sensores.

O meu receio principal, corresponde a que algo aconteça à PCB e não consiga apresentar com qualidade pretendida o meu projeto final de curso (PAP).

4. Referências

Solar Powered WiFi Weather Station V3.0 – projeto do instructables(2021)
<https://www.instructables.com/Solar-Powered-WiFi-Weather-Station-V30/>

Sensor - BME280
<https://www.azde.ly/en/collections/wish/products/gy-bme280>

Informação BME280
<https://lastminuteengineers.com/bme280-arduino-tutorial/>

Informação DS18B20
<https://www.electrofun.pt/sensores-arduino/sensor-temperatura-ds18b20>

significado de LUX
<https://www.merriam-webster.com/dictionary/lux>

Sensor - BH1750
<https://www.joom.com/pt-br/products/5db3564136b54d0101452c57>

Sensor - BH1750
<https://learn.adafruit.com/adafruit-bh1750-ambient-light-sensor>

Documentação do Arduino
<https://docs.arduino.cc/>

Visual Studio Code
<https://code.visualstudio.com/docs>