

Relatório Intermédio

Sistema de Medição de Vibrações

Equipa H



Projeto realizado no âmbito da unidade curricular de
Engenharia de Sistemas



Departamento de Engenharia Eletrotécnica e de
Computadores
Faculdade de Engenharia da Universidade do Porto
Portugal
2022/2023

Índice

1	Desafio	2
2	Lista de requisitos	2
3	System Breakdown Structure	4
4	Arquitetura funcional	6
5	Organização da Equipa	7
5.1	Descrição dos Cargos	7
5.2	Distribuição dos Cargos	7
6	Work Breakdown Structure	9
7	Gestão de Riscos	11
8	Trabalho desenvolvido	13
8.1	Trabalho Realizado	13
8.1.1	PCB sensor	13
8.2	Trabalho a Desenvolver	14
8.2.1	Interface do utilizador	14
8.2.2	Comunicação Raspberry Pi Pico W - InfluxDB	15

1 Desafio

O desafio proposto consiste no desenvolvimento de um sistema capaz de medir as vibrações numa turbina eólica.

Este sistema deve transmitir a informação de modo a atualizar uma base de dados e apresentar os dados numa interface gráfica disponível a qualquer momento para acesso ao utilizador.

2 Lista de requisitos

Nesta secção, estão identificados os requisitos que a equipa definiu com base no pedido do cliente e o que ele considera fundamental. Para este projeto, essas características encontram-se enumeradas abaixo:

- O sistema deverá conseguir registar vibrações com frequências entre os 0.1 Hz e os 100 Hz.
- Três sensores serão necessários: um para medir as vibrações no topo da turbina eólica, outro a meio e o restante na parte inferior.
- As medições efetuadas deverão ser transferidas por Wi-Fi.
- Os dados deverão ser guardados numa base de dados.
- A aplicação deverá apresentar os dados, bem como uma análise básica dos mesmos, preferencialmente por gráficos.

Tendo em conta estas características, os requisitos definidos entre a equipa encontram-se divididos em funcionais (F) e não funcionais (N). Os requisitos funcionais descrevem funções que o sistema deverá implementar; por sua vez, os não funcionais indicam características ou restrições ao modo de funcionamento do sistema. Além disso, conforme o subsistema em que se inserem, cada um dos requisitos é classificado como referente ao subsistema de medição (M), do servidor (S) ou da aplicação (A). Os requisitos definidos entre a equipa estão expostos na tabela 2.

Tabela 1: Requisitos funcionais e não funcionais

Código	Requisito	Prioridade
MF1	Medição da vibração	Alta
MF2	Comunicação entre o módulo Arduino e a Raspberry Pi Pico realizar-se por Wi-Fi	Alta
MN3	O conjunto Arduino + Sensor estar protegido das condições ambientais	Média
SF1	Comunicação entre a Raspberry Pi Pico e a base de dados	Alta
SN2	Recurso ao Telegraf	Média
SN3	Formatação da informação durante comunicação com o servidor	Baixa
SN4	Os dados transmitidos devem ser menos de 5MB/5 minutos	Média
AN1	Aplicação aceder à base de dados	Alta
AF2	Visualização dos dados recorrendo a gráficos	Média
AF3	Análise básica dos dados e respetiva representação	Alta
AN4	Acesso à aplicação através de um <i>web browser</i>	Média

3 System Breakdown Structure

O presente projeto desenvolvido pela equipa visa o desenvolvimento de um produto, que será, fisicamente, uma caixa com sensores integrados capazes de medir vibrações. Esta caixa será depois instalada em turbinas eólicas.

Além disto, ainda uma plataforma web capaz de mostrar ao utilizador algumas métricas pertinentes para monitorizar a turbina.

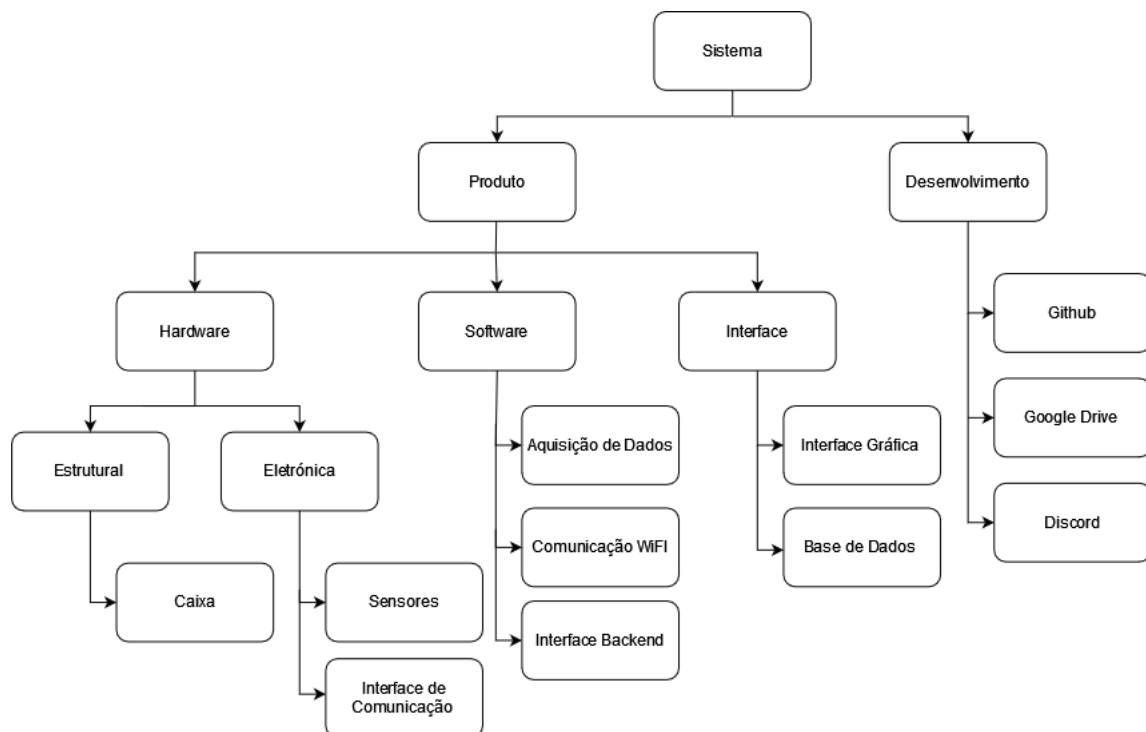


Figura 1: System Breakdown Structure

O nosso sistema está dividido em duas grandes partes: o Produto e o Desenvolvimento.

O subsistema Desenvolvimento é constituído por ferramentas que ajudam à coordenação e evolução do projeto, nomeadamente o *GitHub* para controlar versões de código, *Google Drive* para organização, documentação adicional, partilhada com toda a equipa e o *Discord* para comunicação entre todos os elementos da equipa.

O Subsistema do Produto é constituído por três subsistemas, o *Hardware*, o *Software* e a *Interface*.

O Sub-subsistema de Hardware está, por sua vez, dividido em 2 subsistemas, o sub-sistema Estrutural que representa a parte exterior do nosso produto, a caixa, e o sub-sistema Eletrónica que acomoda os sensores e interfaces de comunicação entre sensor e restantes subsistemas.

Em conjugação com o *Hardware*, existe o Sub-subsistema *Software*, que engloba a aquisição de dados, comunicações *Wifi* com o resto do produto e o *Backend*.

Por fim, a Interface é constituída pela Interface Gráfica e a Base de Dados.

4 Arquitetura funcional

Este modelo arquitetónico tem como principal objetivo identificar a função de sistema e as suas respetivas interações. Por outras palavras, define como as funções funcionarão em conjunto de modo a atingir os objetivos do sistema. Algumas das suas vantagens estão relacionadas com maior eficiência e consistência, melhor desempenho e redução de trabalho a longo prazo.

Na figura seguinte está representada arquitetura funcional deste projeto.

Os módulos principais desta arquitetura, em específico, são o Arduino e a Interface Gráfica. O Arduino é responsável pela aquisição de dados brutos provenientes do sensor de vibração que são posteriormente convertidos em valores de frequência de vibração. A seguir os dados são enviados do Arduino para o Raspberry PI sobre WiFi. O Raspberry PI guarda os dados recebidos numa base de dados.

A interface gráfica permite ao utilizador visualizar os valores de vibração guardados na base de dados, assim como pedir outros valores relevantes, tal como a média, valores máximos e mínimos, entre outros.

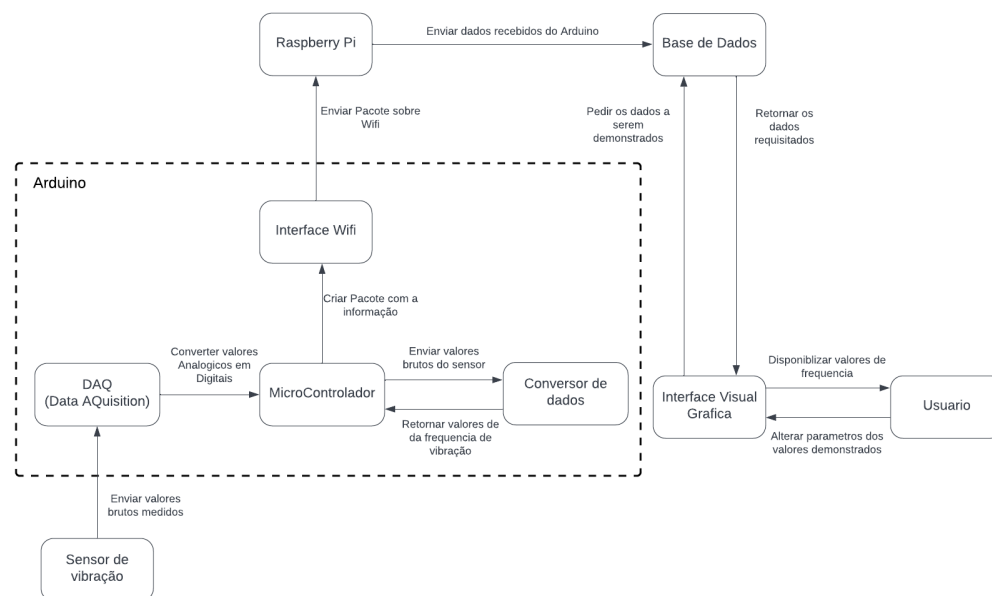


Figura 2: Diagrama de arquitetura funcional

5 Organização da Equipa

Nesta secção, pretende-se clarificar a organização da equipa, mais especificamente os cargos existentes, as respetivas funções, e sua distribuição. Porém, ressalva-se a possível criação de novos cargos no decorrer do trabalho, bem como uma possível redistribuição de cargos se tal for considerado pertinente.

5.1 Descrição dos Cargos

Foram criados cargos dentro da equipa, com responsabilidades bem definidas, por forma a garantir o seu bom funcionamento. Estes cargos estão representados na tabela 1, assim como as responsabilidades a eles associadas.

5.2 Distribuição dos Cargos

Após a criação dos cargos, procedeu-se à sua distribuição entre os elementos da equipa. O seu resultado encontra-se exposto na tabela 2.

Nome	Número Estudante	Cargo atribuído
Catarina Reste	201806184	Secretária
Diogo Andrade	201806693	Sub-líder
Francisco Sousa	201806136	—
Gonçalo Queirós	201806758	Responsável pela documentação
João Freixo	201806867	Responsável pela validação
Pedro Barros	201806365	Facilitador
Ricardo Silva	201809483	—
Sérgio Silva	201806456	Líder
Vitória Correia	201806870	—

Tabela 3: Constituição da equipa e respetivos cargos

Cargo	Funções
Líder	<ul style="list-style-type: none"> • Divisão das tarefas e definição dos prazos de entrega • Distribuição das tarefas pelos membros da equipa • Agendamento das reuniões • Realizar a comunicação com o docente/cliente
Sub-líder	<ul style="list-style-type: none"> • Substituição do líder da equipa na possível eventualidade de este ficar impossibilitado de exercer as suas funções
Secretário	<ul style="list-style-type: none"> • Elaboração das atas das reuniões • Responsável por tomar notas sobre os assuntos discutidos nas reuniões de equipa
Facilitador	<ul style="list-style-type: none"> • Moderação das reuniões semanais entre a equipa • Garantia de que todos os elementos da equipa tenham oportunidade de intervir na sessão
Responsável pela validação	<ul style="list-style-type: none"> • Gestão do repositório <i>GitHub</i> associado ao projeto • Testar e verificar possíveis alterações ao <i>main branch</i> do projeto
Responsável pela documentação	<ul style="list-style-type: none"> • Revisão da escrita da documentação a ser produzida pela equipa • Responsável por garantir a execução de toda a documentação necessária para o projeto

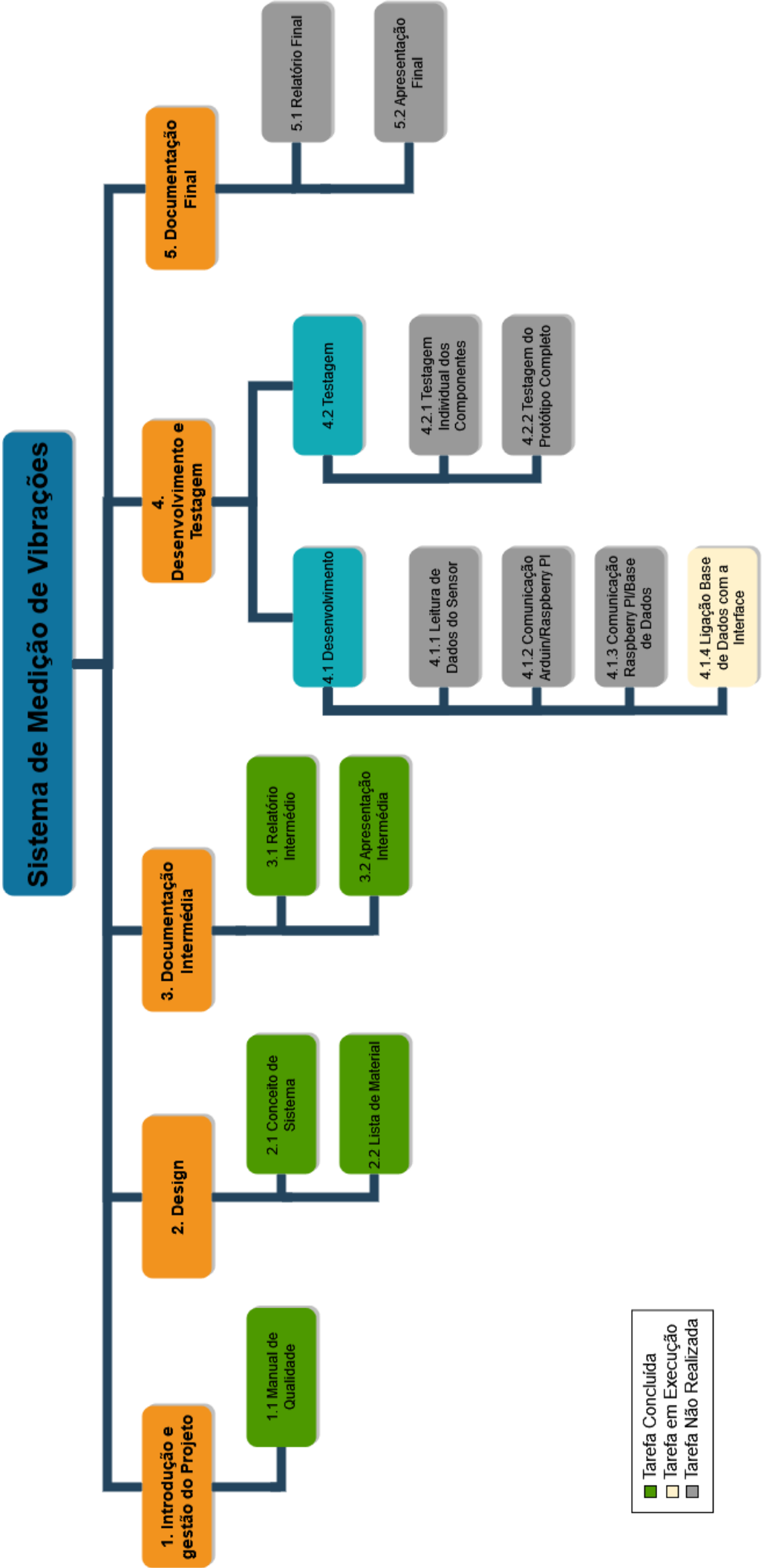
Tabela 2: Cargos criados dentro da equipa

6 Work Breakdown Structure

A Work Breakdown Structure (WBS), também chamada estrutura analítica do projeto, é a divisão do projeto na totalidade em partes mais pequenas e simples de modo a facilitar a gestão do mesmo.

O objetivo desta divisão é identificar os elementos-chave do projeto e definir subequipas responsáveis por um elemento. A partir do momento em que as subequipas são definidas, cabe a cada uma discutir entre si e determinar o que é necessário para atingir o pretendido.

Com o desenvolvimento do projeto, esta estrutura é sujeita a mudanças, possivelmente semanalmente se os chefes de subequipa propuserem e a equipa achar necessário. A estrutura pode ser vista na figura seguinte:



■	Tarefa Concluída
■	Tarefa em Execução
■	Tarefa Não Realizada

7 Gestão de Riscos

Com cada decisão tomada pela equipa e com cada tarefa a executar está associado um risco intrínseco. Deste modo, é expetável que ocorram situações prejudiciais ao normal desenvolvimento do projeto. Tendo isto em conta, é preciso avaliar a probabilidade de ocorrência e a severidade ou impacto que a situação terá na eventualidade de ocorrer.

Após esta análise, devem ser definidas estratégias para mitigar os efeitos que as ocorrências possam ter sobre a equipa e/ou projeto.

Uma análise de algumas destas situações pode ser vista na figura e tabela seguintes:

		IMPACTO		
PROBABILIDADE		ACEITÁVEL	SÉRIO	CRÍTICO
	IMPROVÁVEL			
	POSSÍVEL			
	PROVÁVEL			

Figura 3: Matriz de Risco

Tabela 4: Plano de Gestão de Risco

Causa	Risco	Probabilidade	Impacto	Estratégia
Datas	Atraso na conclusão de uma tarefa	Provável	Aceitável	Rever e redefinir a distribuição de tarefas
	Atraso na entrega do material	Possível	Crítico	Procurar material alternativo emprestado e começar por realizar as tarefas que não dependem de material
Material	Escolha incorreta de método ou solução	Possível	Crítico	Estudar novamente e mais aprofundadamente o tópico
	Material danificado	Improvável	Crítico	Tentar ter material extra ou soluções alternativas
	Falta de conhecimento sobre a tecnologia	Possível	Crítico	Estudar mais sobre a tecnologia a utilizar antes de realizar as tarefas
Equipa	Falta de comunicação entre a equipa	Possível	Aceitável	Aumentar o contacto entre membros da equipa e realização mais frequente de reuniões de equipa e subequipa
	Conflito entre membros da equipa	Improvável	Aceitável	Tratar o assunto entre os membros em questão. Caso isto não seja possível, comunicar o problema ao líder de equipa

8 Trabalho desenvolvido

8.1 Trabalho Realizado

8.1.1 PCB sensor

Seguindo a datasheet do sensor e modo a o utilizar, precisamos de montar o seguinte circuito:

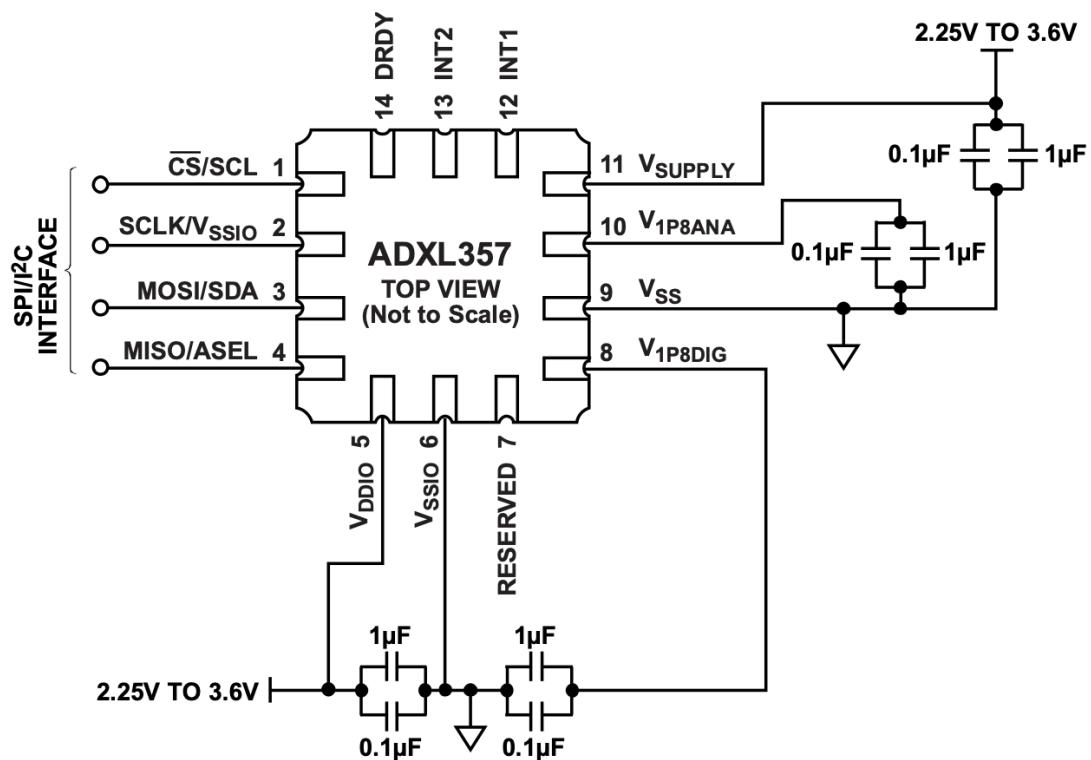


Figura 4: Circuito Sensor

Como não temos limitações de espaço, decidimos desenhar e desenvolver uma pequena *PCB* capaz de funcionar como uma *Pin-out-Board*.

Tem as dimensões necessárias para, numa primeira fase de testes, ser montada numa *BreadBoard*. No futuro será encaixada numa *Perforated PCB board*, assim como os restantes componentes vistos na figura 4.

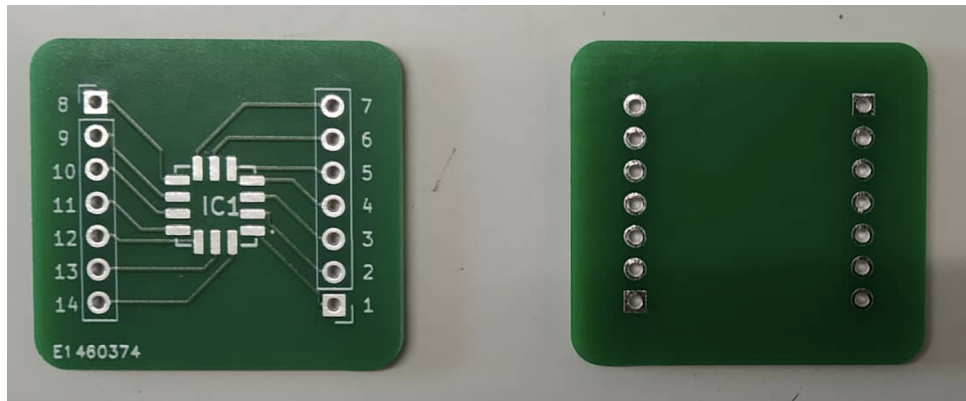


Figura 5: Pin-Out-Board

8.2 Trabalho a Desenvolver

8.2.1 Interface do utilizador

O utilizador terá uma interface a partir da qual terá acesso a métricas relevantes para análise da vibração da turbina eólica.

As métricas serão as seguintes:

- Valor mais recente
- Valor máximo
- Valor mínimo
- Média dos últimos 15 dias
- Valores dos últimos 15 dias

Para mostrar a informação ao utilizador, vamos utilizar:

- Gráficos medidores
- Gráficos de linha
- Etiquetas de texto



Figura 6: Exemplo Interface Utilizador

8.2.2 Comunicação Raspberry Pi Pico W - InfluxDB

De forma a estabelecer a ligação entre o *Raspberry Pi* e a base de dados será necessária a instalação do *software Telegraf* de recolha de dados. Para tal iremos, inicialmente, prepara a instalação do *Influx* no *Raspberry*:

```
wget -qO- https://repos.influxdata.com/influxdb.key | sudo  
tee /etc/apt/sources.list.d/influxdb.list test $VERSION_ID  
= "8" &&  
echo "deb https://repos.influxdata.com/debian jessie stable"  
| sudo tee /etc/apt/sources.list.d/influxdb.list test $VERSION_ID  
= "9" &&  
echo "deb https://repos.influxdata.com/debian stretch stable"  
| sudo tee /etc/apt/sources.list.d/influxdb.list
```


E executá-la:

```
$ sudo apt-get update && sudo apt-get install influxdb
```

Iniciando o serviço e verificando o seu estado:

```
$ sudo service influxdb start  
$ sudo service influxdb start
```

Estando a funcionar passaremos a instalar o *Telegraf*:

```
$ wget https://dl.influxdata.com/telegraf/releases/telegraf_1.12.2-1_armhf.deb  
$ sudo dpkg -i telegraf_1.12.2-1_armhf.deb
```

Correndo posteriormente um *script* de recolha de dados, por exemplo:

```
$ nohup python3 data_col_script.py &
```
