

Faculdade de Engenharia da Universidade do Porto



**Desenvolvimento de um Sistema sem Fios para a
Monitorização e Alerta de Sinais Vitais em Animais
de Companhia em Internamento Hospitalar**

Diana Patrícia Gomes Duarte

setembro de 2020

Faculdade de Engenharia da Universidade do Porto



**Desenvolvimento de um Sistema sem Fios para a
Monitorização e Alerta de Sinais Vitais em Animais
de Companhia em Internamento Hospitalar**

Diana Patrícia Gomes Duarte

Dissertação realizada no âmbito do
Mestrado em Engenharia Biomédica

Orientador: Professor Doutor Joaquim Gabriel Magalhães Mendes (FEUP)
Coorientador: Professor Doutor Luís Miguel Joaquim Marques Antunes (UTAD)

setembro de 2020

© Diana Patrícia Gomes Duarte, 2020

“O impossível é quase sempre o que nunca se tentou.”

Elie Wiesel

Resumo

A relação entre os animais e os seres humanos existe desde os tempos antigos, mas ao longo dos anos tem sofrido alterações. No passado, os animais eram adquiridos apenas como ferramenta de trabalho, razões militares, ou meio de transporte. Hoje eles são principalmente animais de companhia/estimação. Uma vez que, eles não têm a capacidade de se expressar, é necessário criar sistemas de monitorização e controlo para que os donos e centros clínicos, tenham uma melhor percepção do seu estado de saúde.

Ao longo dos anos, foram surgindo vários equipamentos para a medição automatizada dos sinais fisiológicos dos animais. Inicialmente estes equipamentos baseavam-se apenas em chips e monitores conectados através de cabos. No entanto, com a maior integração da eletrónica e a vulgarização das comunicações Bluetooth e Wi-Fi começaram a surgir os primeiros coletes e coleiras de monitorização animal. O grande problema dos dispositivos desenvolvidos recai no seu custo e dificuldades de adaptação destes dispositivos às diferentes espécies.

A monitorização contínua de sinais vitais durante a fase de recobro pós-cirúrgico é de grande importância, uma vez que é quando o animal recupera todos os seus sinais fisiológicos retirados pela anestesia. O mesmo se aplica a animais em internamento e cuidados intensivos que podem apresentar alterações da sua atividade. Os equipamentos desenvolvidos para estes períodos ainda são escassos, exigindo assim que haja a presença constante de um médico ou enfermeiro junto do animal.

Nesta dissertação foi desenvolvido um sistema de monitorização para animais de companhia, para acompanhar a fase de recobro pós-cirúrgico e internamento. Este sistema utiliza diversos sensores e comunicação Wi-Fi para medir diferentes parâmetros fisiológicos e comportamentais do animal. Desta forma deseja-se vigiar essa monitorização à distância, enviar os valores para uma base de dados e analisá-los posteriormente num site.

O sistema de monitorização desenvolvido baseou-se na utilização de diversos sensores para a medição de parâmetros como o movimento, temperatura, quantidade de comida e bebida ingerida e temperatura ambiente e humidade. Estes, em conjunto com microcontroladores como o Arduino UNO e módulos Wi-Fi ESP8266, permitiram que os valores medidos estivessem constantemente a ser enviados para o computador, de forma totalmente remota. Posteriormente, a parte de software consistiu na criação de uma base de dados, para onde os valores recolhidos vão ser enviados e guardados, e um site, onde a análise da monitorização será apresentada graficamente.

O sistema desenvolvido respondeu a todos os pedidos e permitiu transmitir ao corpo clínico as informações necessárias do animal sem causar qualquer tipo de interferência no seu bem-estar.

Palavras-chave: sinais vitais, animal, monitorização à distância, comunicação Wi-Fi.

Abstract

The relationship between animals and humans has existed since ancient times, but over the years it has changed. In the past, animals were acquired only as a work tool, military reasons, or as a means of transport. Today they are mostly companion/pet animals. Since they do not have the capacity to express themselves, it is necessary to create monitoring and control systems so that owners and clinical centers have a better perception of their health status.

Over the years, various equipment has emerged for the automated measurement of the physiological signals of animals. Initially, this equipment was based only on chips and monitors connected by cables. However, with the greater integration of electronics and the popularization of Bluetooth and Wi-Fi communications, the first animal monitoring vests and collars began to appear. The big problem with the developed devices lies in their cost and difficulties in adapting these devices to different species.

Continuous monitoring of vital signs during the post-surgical recovery phase is of great importance, since it is when the animal recovers all of its physiological signs removed by anesthesia. The same applies to animals in hospital and intensive care that may show changes in their activity. The equipment developed for these periods is still scarce, thus requiring the constant presence of a doctor or nurse with the animal.

In this dissertation, a monitoring system for companion animals was developed to monitor the post-surgical recovery and hospitalization phase. This system uses several sensors and Wi-Fi communication to measure different physiological and behavioral parameters of the animal. In this way, we want to monitor this monitoring from a distance, send the values to a database and analyze them later on a website.

The monitoring system developed was based on the use of several sensors to measure parameters such as movement, temperature, amount of food and drink ingested and room temperature and humidity. These, together with microcontrollers such as Arduino UNO and Wi-Fi modules ESP8266, allowed the measured values to be constantly being sent to the computer, completely remotely. Subsequently, the software part consisted of the creation of a database, where the collected values will be sent and saved, and a website, where the monitoring analysis will be presented graphically.

The developed system responded to all requests and allowed to transmit to the clinical staff the necessary information of the animal without causing any type of interference in its well-being.

Keywords: Vital Signs, Animal, Remote Monitoring, Wi-Fi Communication.

Agradecimentos

Começo por agradecer à minha família, especialmente aos meus pais, por todo o apoio ao longo de todo o meu percurso, por nunca me deixarem desistir e por acreditarem sempre no meu sucesso.

Aos meus melhores amigos e companheiros de universidade, um obrigada especial pela paciência, por estarem sempre ao meu lado, por me incentivarem a ir mais longe e por me fazerem acreditar que é possível.

Quero agradecer aos meus orientadores, Joaquim Gabriel e Luís Antunes, por toda a disponibilidade na orientação deste estudo, pela ajuda e principalmente pela paciência. Sem eles a conclusão deste trabalho não era possível.

Por fim, um agradecimento especial ao i3s (instituto de investigação e inovação em saúde) ao HVUTAD (Hospital veterinário da Universidade de Trás-os-Montes e Alto Douro) e principalmente ao hospital veterinário AnimalCare no Porto por estarem sempre disponíveis para me receber e ajudar a perceber o que realmente era necessário mudar na medicina veterinária. Com a vossa opinião o protótipo teve uma voz experiente.

Índice

Resumo	v
Abstract.....	vii
Agradecimentos	ix
Lista de figuras.....	xiv
Lista de tabelas	xvii
Abreviaturas, Acrónimos e Símbolos	xix
1 Introdução	1
1.1 - Cuidados Hospitalares.....	2
1.2 - Monitorização animal pós-cirúrgica.....	4
1.2.1 - Monitorização cardiovascular	5
1.2.2 - Monitorização respiratória	6
1.2.3 - Monitorização da temperatura corporal	6
1.2.4 - Monitorização da dor	6
1.3 - Motivação	7
1.4 - Objetivos	7
1.5 - Estrutura da dissertação.....	8
2 Estado da arte.....	9
2.1 - Necessidade hospitalar	12
3 Protótipo a desenvolver	17
3.1 - Arduíno UNO	18
3.1.1 - Arduíno IDE.....	19
3.2 - Placa de desenvolvimento ESP8266 NodeMCU	20
3.3 - WeMos D1 mini NodeMcu - ESP8266	21
3.4 - Sensor de temperatura e humidade DHT22	22
3.4.1 - Programação no arduíno IDE	22
3.5 - Sensor de temperatura IR MLX90614	24
3.5.1 - Programação no arduíno IDE	25
3.6 - Câmara LS_Y201.....	26

3.6.1 - Programação no arduino IDE	27
3.7 - Células de carga.....	29
3.7.1 - Primeira calibração	30
4 Implementação do protótipo	33
4.1 - Montagem da jaula	33
4.2 - Programação em arduino IDE	35
4.2.1 - Células de carga - NodeMcu ESP8266-E12	35
4.2.2 - Sensores - Arduino UNO - WeMos D1 mini ESP8266	36
4.3 - Base de dados	37
4.3 - Comunicação base de dados - site	39
4.4 - Site ANIMALcare.....	40
5 Resultados	45
6 Conclusão	51
6.1 - Custos do projeto	52
6.2 - Trabalhos futuros	52
Referências	55
Anexo A	61
Anexo B - Esquemáticos	63
B1 - células de carga - ESP8266 NodeMCU	63
B2 - DHT22, MLX90614 - Arduino UNO - WeMos D1 mini	64
Anexo C - Código arduino.....	65
C1 - células de carga - ESP8266 NodeMCU	65
C2 - DHT22, MLX90614 - Arduino UNO - WeMos D1 mini	70
Anexo D - código da base de dados.....	74

Lista de figuras

Figura 1.1 - Sala de Internamento do Hospital ANIMALcare do Porto.	3
Figura 1.2 - Monitorização animal durante a cirurgia (A) e durante a fase de internamento (B) (fotografias cedidas pelo hospital ANIMALcare).	5
Figura 2.1 - Análise gráfica da resposta à segunda questão do inquérito: “Quais considera os maiores problemas na monitorização, na fase de recobro?”	13
Figura 2.2 - Análise gráfica da resposta à terceira questão do inquérito: “Na fase de recobro, quais os parâmetros mais importantes a monitorizar?”	14
Figura 2.3 - Análise gráfica da resposta à sétima questão do inquérito: “Implementando um sistema destes, quais as espécies em que prevê maior utilização?”.	15
Figura 3.1 - Desenho de arquitetura.	18
Figura 3.2 - Arduino UNO [59].	19
Figura 3.3 - Layout do arduino IDE.	20
Figura 3.4 - NodeMCU ESP8266-ESP12.	21
Figura 3.5 - WeMos D1 mini NodeMcu - ESP8266.	21
Figura 3.6 - Sensor DHT22.	22
Figura 3.7 - Conexão do sensor DHT22 ao arduino UNO.	23
Figura 3.8 - Instalação da biblioteca do DHT22.	23
Figura 3.9 - Código de programação para o sensor DHT22.	24
Figura 3.10 - Sensor de temperatura IR MLX90614.	25
Figura 3.11 - Conexão do MLX90614 ao arduino UNO.	25
Figura 3.12 - Código de programação para o sensor MLX90614.	26
Figura 3.13 - Câmara LS-Y201.	26
Figura 3.14 - Conexão da câmara LS-Y201 ao arduino UNO.	27
Figura 3.15 - Código de programação para a câmara LS-Y201.	27
Figura 3.16 - Diagrama da célula de carga[67].	29
Figura 3.17 - Amplificador HX711.	30
Figura 3.18 - Esquema de ligação entre a célula de carga, o amplificador HX711 e o Arduino UNO [68].	30

Figura 3.19 - Código de programação para a célula de carga.	31
Figura 4.1 - Jaula construída com os sensores colocados.	34
Figura 4.2 - Diagrama UML da base de dados animalcare.	38
Figura 4.3 - Diagrama de fluxo.	39
Figura 4.4 - Funções disponíveis no cabeçalho do site.	41
Figura 4.5 - Funcionalidade do site, disponíveis para o médico.	43
Figura 4.6 - Funcionalidades disponíveis no site quando se entra como tutor.	44
Figura 5.1 - Monitor de visualização dos valores dos sensores.	46
Figura 5.2 - Gráfico obtido da temperatura animal (linha verde), da temperatura ambiente (linha castanha) e da humidade relativa (linha laranja).	46
Figura 5.3 - Gráfico obtido pelo movimento do animal.	47
Figura 5.4 - Gráfico obtido pela célula de carga colocada no bebedouro do animal.	47
Figura 5.5 - Gráfico obtido pela célula de carga colocada no comedouro do animal.	48
Figura 5.6 - Correlação entre os diferentes gráficos para mostrar momentos coincidentes.	49

Lista de tabelas

Tabela 1.1 - Sinais vitais mais importantes a avaliar [19].	5
Tabela 2.1 - Informações acerca de alguns sistemas de monitorização.	10
Tabela 2.2 - Respostas recolhidas sobre os equipamentos existentes para a monitorização e as limitações que estes apresentam	14
Tabela 2.3 - Vantagens do sistema recolhidas pela opinião de clínicos.....	15
Tabela 2.4 - Sugestões enunciadas pelos clínicos como resposta à última questão do inquérito: “Caso queira deixar alguma sugestão ou informação que lhe tenha surgido ao longo deste questionário, no sentido de melhorar a ideia ou até mesmo o dispositivo, coloque aqui.”.....	15
Tabela 4.1 - Bibliotecas necessárias para a conexão da célula de carga com o NodeMCU ESP8266-E12.	35
Tabela 4.2 - Bibliotecas necessárias para a conexão do DHT22 e do MLX90614 com o arduíno UNO e o WeMos D1 mini ESP8266.	37
Tabela 6.1 - Preço dos componentes adquiridos.	52

Abreviaturas, Acrónimos e Símbolos

Lista de abreviaturas e acrónimos (ordenadas por ordem alfabética)

ADC	Analogic Digital Conversor
AVC	Acidente vascular cerebral
bpm	Batimentos por minuto
CSS	Cascading Style Sheets
DSP	Digital Signal Processor
ECG	Eletrocardiograma
FC	Frequência cardíaca
FR	Frequência respiratória
GND	Ground
GPS	Sistema de posicionamento global
HTML	HyperText Markup Language
HTTP	Hypertext Transfer Protocol
HVUTAD	Hospital veterinário da Universidade de Trás-os-Montes e Alto Douro
IDE	Integrated Development Environment
IoT	<i>Internet of Things</i>
IR	Infrared
I2c	Inter integrated Circuit
i3s	Instituto de investigação e inovação em saúde
JPEG	Joint Photographic Experts Group
LED	Light emitting diode
MDF	Medium Density Fiberboard
PA	Pressão arterial
PHP	Hypertext Preprocessor
PWM	Pulse wide modulation
RST	Reset
RX	Receive data
SCK	Serial clock
SCL	System Control Language
SDA	Serial data

SMBUS	System Management BUS
SPI	Serial Peripheral Interface
SQL	Structured Query Language
T	Temperatura
TX	Transmit data
TCP/IP	Transmission Control Protocol/Internet Protocol
UART	Universal asynchronous receiver transmitter
UML	Unified Modeling Language
USB	Universal Serial Bus
VDD	Power supply pin

Lista de símbolos

g	Grama
h	hora
m	metro
m	minuto
mA	miliampere
MB	Megabyte
ml	mililitro
mm	milimetro
mmHg	milimetro de mercúrio
mV	milivolt
°C	graus Celsius
O ₂	Oxigénio
CO ₂	Dióxido de carbono
KB	Kilobyte
Kg	Quilograma
V	Volt
%	Porcentagem
€	Euro

Capítulo 1

Introdução

A relação entre os animais e os seres humanos existe desde os tempos antigos, mas ao longo dos anos tem sofrido alterações. No passado, os animais eram adquiridos apenas como ferramenta de trabalho, razões militares, ou meio de transporte, consoante as necessidades. Contudo, hoje em dia, eles são principalmente animais de companhia/estimação [1, 2]. A aquisição de um animal envolve muitos benefícios, como a diminuição da solidão, a promoção da interação entre os donos e outras pessoas, experiências de humor e brincadeira, sentimentos de fidelidade, melhoria na saúde física e mental e principalmente a diversão em casa. Estas vantagens não são verificadas apenas nos adultos, mas também nas crianças. As crianças na presença de um animal apresentam um maior sentido de segurança, melhor desenvolvimento da linguagem e adquirem um sistema imunológico mais fortalecido [3].

Os aspetos positivos de ter animais domésticos são difíceis de quantificar, mas são certamente relevantes, o que faz com que estes sejam encarados como membros da família [3]. Dada esta importância, é necessário quando se adquire um animal, assumir toda a responsabilidade sobre ele e prestar-lhe todos os cuidados de alimentação, saúde e bem-estar. Para tal, e tendo em atenção o facto de estes não terem a capacidade de se expressar, é necessário criar sistemas de monitorização e controlo para que os donos e até mesmo os centros clínicos, tenham uma melhor percepção do estado de saúde dos animais.

Ao longo dos anos foram surgindo vários equipamentos para a medição automatizada, principalmente dos sinais fisiológicos dos animais. Inicialmente, começaram apenas a ser comercializados monitores para as variáveis cardíaca e respiratória durante a cirurgia, ajudando no controlo da anestesia [4]. Mais recentemente, os sistemas evoluíram, incluindo comunicação sem fio, Bluetooth ou Wi-Fi. Esta evolução permite não só a monitorização em contexto clínico, mas também o controlo da saúde do animal nos seus lares, através de aplicações utilizadas pelos tutores. Os primeiros desenvolvimentos neste contexto foram dirigidos para o controlo da localização do animal através de chips ou coleiras GPS [5]. No entanto, essas tecnologias evoluíram para sistemas que detalham o estado de saúde do animal.

Hoje em dia, existem diversos tipos de coleiras e coleiras que monitorizam os sinais vitais do animal, desde o batimento cardíaco, temperatura, nível de oxigénio, assim como outros tipos de parâmetros, como as calorias consumidas e distâncias percorridas. Os dados recolhidos

por estes monitores, são enviados para um dispositivo móvel através de uma simples aplicação. Em sistemas mais desenvolvidos a informação pode ser transmitida para dois locais distintos, o centro veterinário e o dispositivo móvel do tutor do animal [6]. A transmissão deste tipo de informações para os centros veterinários permite resolver vários problemas, como o tempo despendido nos exames físicos e diagnóstico, diminuindo bastante o manuseamento do animal, e obtendo resultados mais rápidos [7]. Em contrapartida, estes sistemas não são muito acessíveis, o que faz com que ainda não sejam muito utilizados em centros veterinários.

Um ponto negativo, que pode ser crucial para o animal, é o facto de sistemas de monitorização deste tipo ainda não estarem desenvolvidos para o acompanhamento pós-cirúrgico. Sinais vitais como a pressão arterial (PA), temperatura (T), frequência cardíaca (FC) e respiratória (FR) evidenciam alterações da função corporal e são indispensáveis na monitorização da saúde animal. Estas informações devem ser controladas quer os animais estejam sob efeito de anestesia ou não [8]. No entanto, é durante o processo anestésico e consequente recuperação que a monitorização assume uma importância maior na sobrevivência do animal no contexto clínico, uma vez que a anestesia provoca um défice dos sentidos e movimentos musculares. Dispositivos e sistemas de monitorização neste ambiente permitem detetar e alertar eventuais alterações fisiológicas. Desta forma, situações que possam comprometer a vida do paciente, podem ser rapidamente corrigidas durante a cirurgia e na fase de recobro e internamento [9]. Esta particularidade permite um maior conforto não só para o animal como para toda a equipa clínica, uma vez que deixaria de ser necessário o controle presencial. Situações e necessidades idênticas estão identificadas para animais sujeitos a alguns internamentos e cuidados intensivos.

1.1 - Cuidados Hospitalares

Tal como nos humanos, a esperança média de vida dos animais tem vindo a aumentar com o desenvolvimento dos cuidados veterinários. Esta evolução foca-se não só no tratamento em si, mas também nas implicações que este tem para o bem-estar do animal [10]. Normalmente, na medicina veterinária, o conceito de bem-estar é, na maioria das vezes, identificado pelo estado de saúde; no entanto, é importante distinguir o sucesso do tratamento a nível da saúde e o bem-estar geral do animal, durante e após os cuidados prestados [11].

Na prática veterinária, há três pontos que devem ser tidos em consideração por parte dos clínicos: bem-estar físico, mental e natural. O primeiro está relacionado com a condição corporal do animal, o seu funcionamento biológico, que vai refletir-se nas doenças, no estado nutricional e nos cuidados que ele necessita [12]. O bem-estar mental é referente à vida psicológica, ou seja, como o animal está a reagir aos processos, as suas capacidades cognitivas e os sentimentos que este expressa. O mais comum, em ambiente hospitalar, é a prevalência de emoções negativas como medo, tristeza, aflição, que afetam a saúde mental e acabam por ter influência na saúde física [12]. Por último, mas não menos importante, o bem-estar natural do animal. Este está relacionado com a sua vida habitual, com a oportunidade que este tem de expressar o seu comportamento, o que nem sempre é possível na totalidade, principalmente em casos de internamento devido a toda a logística do espaço a este disponibilizado (figura 1.1) [12].



Figura 1.1 - Sala de Internamento do Hospital ANIMALcare do Porto.

Dada a relevância dos animais na vida humana, há cada vez mais um maior investimento na melhoria dos cuidados prestados. Os clínicos preocupam-se não só com o bem-estar referido anteriormente, mas também com o avanço e melhoria nos serviços disponibilizados. Hoje em dia, num só centro clínico há um leque enorme de atividades e cuidados que podem ser prestados para que os animais tenham uma boa saúde. Podemos encontrar serviços como banhos e tosquiagens, oftalmologia veterinária, dermatologia, diálise, cirurgia, cardiologia e exames eco cardiográficos, tomografia, imagiologia, anestesia, ortopedia e neurologia, endoscopia, nutrição e até mesmo serviços de acupuntura e fúnebres [13]. O crescimento dos tratamentos tem sido exponencial e permite aos clínicos prestarem, cada vez mais, um melhor e mais viável serviço.

Os animais, mesmo com todos os cuidados recebidos, necessitam, frequentemente de ser submetidos a intervenções cirúrgicas. Este serviço leva, na maioria das vezes, ao internamento do animal por um determinado período que o veterinário define como necessário. Todo o processo, desde a chegada do animal ao centro clínico até à saída e total recuperação deste, deve ter uma atenção redobrada e um controlo cuidadoso. Este pode ser considerado dos procedimentos mais importantes e minuciosos na vida de um animal. Até chegar o momento da cirurgia, há cuidados a ter, sendo o mais importante a indução da anestesia. A anestesia causa inconsciência, ausência de dor e o relaxamento muscular necessário para o médico intervir sem problemas [14]. A ausência de consciência e movimento pode trazer problemas se a recuperação do animal não for devidamente vigiada, podendo levar a danos permanentes ou morte.

Atualmente tem sido sugerida a importância dos cuidados pós-cirúrgicos (é um dos períodos mais importantes para a vida do animal enquanto este se encontra no centro clínico), uma vez que é necessária uma monitorização muito atenciosa, mas ainda assim os estudos sobre estes são ainda escassos [15]. É necessário estar familiarizado com a importância da monitorização pós-cirúrgica para prestar melhores cuidados e evitar complicações que possam ser fatais para o animal.

1.2 - Monitorização animal pós-cirúrgica

A monitorização, definida como a ação de vigiar, observar ou verificar, é algo que deve ser realizado em todos os pacientes, quer estes estejam sob efeitos de anestesia ou não [16]. Numa situação de internamento, principalmente na fase de recobro pós-cirúrgico, esta é essencial para relatar a condição clínica do paciente. Como já referido anteriormente, a anestesia fornece as condições fisiológicas necessárias para o médico proceder à cirurgia, no entanto, no final quando é retirado o suporte de oxigénio e com a ausência e recuperação da atividade fisiológica, podem haver complicações devido ao efeito residual dos agentes anestésicos. Desta forma é crucial a recolha de informação através de uma monitorização vigilante para acompanhar a recuperação plena do animal [17].

No recobro pós-cirúrgico há a fase imediata, onde o foco é a recuperação anestésica, a monitorização e suporte das funções orgânicas do animal e a menos imediata onde é necessário continuar a garantir uma recuperação e cicatrização adequada, prevenindo complicações comuns [18]. Estas fases são cruciais para a recuperação do animal. Durante as mesmas, é necessário garantir o seu bem-estar e limpeza, para prevenir infeções. A monitorização dos sinais vitais e do local da incisão, reabilitação, nutrição e uns bons cuidados de medicina interna são ferramentas essenciais [17, 18].

A monitorização na fase de recobro deve ser assegurada da mesma forma que é durante a cirurgia e apoiando-se não só em exames físicos, mas principalmente nos sinais vitais recolhidos (figura 1.2, 1.3). A tabela 1.1 apresenta as variáveis fisiológicas mais importantes a monitorizar [19]. A monitorização deve ser especialmente rigorosa nas primeiras 3 horas após a cirurgia, uma vez que a maioria das complicações ocorrem neste período [20]. Quanto à frequência da monitorização, esta deve ser estabelecida consoante o estado do animal, mas estando em internamento, o ideal é que seja realizada em curtos períodos de tempo. Para este controlo é fundamental a presença de especialistas veterinários que deverão garantir a transferência de informação pertinente, sempre que necessário, sendo isto facilitado pelo registo numa ficha que posteriormente possibilita também a avaliação da recuperação do animal [21]. Os procedimentos aplicados nesta fase são idênticos aos usados em animais internados em cuidados intensivos.



Figura 1.2 - Monitorização animal durante a cirurgia (A) e durante a fase de internamento (B) (fotografias cedidas pelo hospital ANIMALcare).

Tabela 1.1 - Sinais vitais mais importantes a avaliar [19].

SINAIS VITAIS MAIS IMPORTANTES A MONITORIZAR	
• Frequência e ritmo cardíaco • Padrão e frequência respiratória (FR) • Pulso arterial (intensidade e ritmo)	• Comportamento e postura • Temperatura • Dor

1.2.1 - Monitorização cardiovascular

A monitorização circulatória e cardíaca pode ser medida através de exames físicos, por eletrocardiografia (ECG) e pela pressão arterial sanguínea. A frequência cardíaca (FC) varia conforme a espécie e o porte do animal. No caso dos cães, esta pode variar entre 80-140 batimentos por minuto (bpm) nos animais de pequeno porte e entre 60-120 bpm nos de grande porte [22]. Ao fazer a medição da FC pelo exame físico, esta deve ser medida juntamente com o ritmo cardíaco por auscultação cardíaca, uma vez que vai permitir a deteção de eventuais sopros e até mesmo arritmias [23]. Quanto à medição por ECG, este permite uma verificação da FC e do ritmo cardíaco de uma forma contínua sendo especialmente recomendado para casos de arritmias, bradicardias ou pulso irregular [21, 23]. Por último, no caso de a medição ser feita pela pressão arterial sanguínea, compreende a pressão arterial sistólica, a arterial média e a arterial diastólica. Os valores normais em cães são 120-150 mmHg, 90-110 mmHg e 80-90 mmHg respetivamente, no entanto, estes valores variam com a raça [24].

Um controlo cuidadoso desta monitorização permite evitar problemas como a hipotensão (complicação anestésica mais comum [25]) que compromete a perfusão renal e a circulação sanguínea, hipertensão, arritmias e até mesmo acidentes vasculares cerebrais (AVC).

1.2.2 - Monitorização respiratória

Esta monitorização é principalmente importante em animais com doenças respiratórias ou cardíacas, no entanto não deixa de ser importante nos restantes. A função respiratória pode ser controlada pelo padrão da frequência respiratória, por oximetria de pulso e por gasimetria. A auscultação pulmonar permite revelar ruídos respiratórios, que podem indicar problemas como a existência de líquido na árvore respiratória, taquipneia e dispneia [21, 26]. O oxímetro de pulso mede a quantidade de oxigénio no sangue, porém o seu uso é muito limitado no pós-cirúrgico pois apresenta alguns erros na medição em animais conscientes [27]. Por último, a gasimetria, permite avaliar os gases arteriais. O sangue é recolhido a partir de qualquer artéria superficial e através da concentração da hemoglobina é possível perceber o valor dos gases arteriais e analisar a situação respiratória [28].

A análise dos valores da monitorização permite prevenir e detetar situações de hipoxemia e hipercapnia, situações que podem alertar para problemas na função respiratória e na dificuldade de eliminação do anestésico.

1.2.3 - Monitorização da temperatura corporal

A monitorização da temperatura corporal é dos parâmetros mais importantes, uma vez que os anestésicos alteram os mecanismos de termo regulação [29]. Este parâmetro deve ser medido após o término da cirurgia, sendo que, nas primeiras horas deve ser controlado em intervalos de 10-15 minutos para avaliar toda a evolução [29]. O local mais frequente é a temperatura retal, uma vez que nos dá o valor exato da temperatura corporal, ao contrário da temperatura medida nas extremidades.

No organismo todos os processos enzimáticos ocorrem devidamente se a temperatura corporal estiver entre 37,8 °C - 39,2 °C [29], fora deste intervalo podem surgir várias complicações pós-anestésicas que não melhoram enquanto a temperatura não for regularizada. A monitorização da temperatura no pós-cirúrgico pode ser crucial para o animal. A hipotermia pode conduzir à redução do metabolismo e outros problemas consequentes como hipoventilação, pneumonias e morte, daí ser tão importante a sua prevenção [29, 30]. No caso da hipertermia, apesar de esta ser menos comum, pode ter complicações como convulsões, que levam a um aumento no consumo de oxigénio e causam complicações na recuperação do animal.

1.2.4 - Monitorização da dor

A dor é um parâmetro muito subjetivo e como tal, muito difícil de quantificar. Na fase de recobro pós-cirúrgico a dor é aguda, caracterizada por uma resposta neuro-endócrina que pode levar a vômitos, arritmias, entre outros, e se não for detetada pode mesmo causar morbilidade e mortalidade nesta fase [31]. Alguns dos sinais fisiológicos que podem demonstrar dor é a taquicardia, o aumento da pressão arterial e da temperatura corporal. Dentro dos sinais comportamentais, o normal é o animal revelar muita agitação, agressão, medo, postura anormal, tremer ou até mesmo a alteração da expressão facial [31]. De forma a diminuir a sua subjetividade e melhorar a avaliação da dor o ideal é durante o pós-cirúrgico, em intervalos pequenos, a mesma pessoa ver o animal, de modo a detetar movimentos diferentes ao longo dessa fase.

1.3 - Motivação

A monitorização de sinais vitais em animais tem sido uma área em desenvolvimento e com relativo sucesso em inúmeras situações, principalmente ao nível da possibilidade de controle do animal por parte do seu tutor. O desenvolvimento nesta área tem sido exponencial, nomeadamente através da introdução das tecnologias com comunicação sem fios. Contudo, muitas das soluções encontram-se restritas a certos parâmetros e o seu valor monetário leva a que ainda não tenham o uso necessário em ambiente clínico.

A monitorização do animal em período pós anestésico é crucial, devido à perda dos sentidos por efeito da anestesia. Contudo, ainda não é dada a devida importância à automatização desta monitorização. A monitorização à distância através de sensores e comunicações sem fios que permitam o controle e alerta de anomalias em tempo real, deverá permitir melhorar a qualidade do internamento do animal e facilitar o trabalho clínico.

O forte crescimento económico do mercado dos animais de estimação a nível global constitui outra forte motivação. Este terá superado em 2019, em Portugal, a barreira dos 200 milhões de euros [32].

1.4 - Objetivos

Em concordância com o tema apresentado “Desenvolvimento de um Sistema sem fios para a monitorização e alerta de sinais vitais em animais de companhia em internamento hospitalar”, a presente dissertação tem como objetivo estudar e compreender as necessidades encontradas atualmente nos hospitais veterinários, focando na vertente do internamento pós-cirúrgico. O conhecimento assimilado ao longo deste trabalho tem impacto não só no trabalho do corpo clínico, mas principalmente no bem-estar e tratamento fornecido ao animal.

O principal objetivo desta dissertação é implementar um sistema de monitorização para animais de companhia, cães e gatos, para a fase de recobro pós-cirúrgico e internamento. Para o efeito, utilizar-se-ão sensores como células de carga para o controle de movimento do animal e o controlo da quantidade de comida e bebida ingerida. Uma câmara para vigília em tempo real, onde a utilização desta permitirá também aos tutores “visitar” o animal remotamente, aumentando a comodidade para ambos os lados. Por fim, será ainda implementada uma câmara térmica para a medição da temperatura corporal do animal. Será implementado ainda, um sensor de temperatura e humidade, para que haja não só o controle do paciente, mas também do ambiente que o rodeia.

Os sensores e câmaras aplicados na jaula vão permitir a recolha e controle de dados sobre o paciente, em tempo real. A partir da informação recolhida, o sistema pode alertar o médico de situações de anomalia, e consequentemente, aumentar a margem de segurança do animal e diminuir a mortalidade e morbilidade durante esta fase crucial.

1.5 - Estrutura da dissertação

A dissertação está organizada em seis capítulos diferentes. Cada capítulo tem como finalidade permitir ao leitor uma mais fácil compreensão do tema, tendo em vista as diversas componentes para a sua realização.

- **Capítulo 1 - Introdução:** apresenta-se o problema, uma descrição dos cuidados hospitalares prestados atualmente nas clínicas veterinárias e a importância da monitorização pós-cirúrgica e internamento de animais, especificando as medidas tomadas em relação aos parâmetros mais importantes de monitorizar nesta fase. Aborda, ainda, a principal motivação e os objetivos propostos.

- **Capítulo 2 - Estado da arte:** apresenta uma revisão dos sistemas e dispositivos de monitorização existentes no mercado, apresentando vantagens e desvantagens dos mesmos, e em que vertentes da veterinária se aplicam mais. É feita, também, uma análise dos inquéritos realizados a vários médicos veterinários, com o intuito de entender o que realmente é necessário em meios hospitalares.

- **Capítulo 3 - Protótipo a desenvolver:** explica o protótipo pretendido para o sistema a desenvolver. Refere todos os parâmetros escolhidos para a monitorização animal e os tipos de comunicações utilizados. Apresenta, ainda, a lista de material escolhido para o seu desenvolvimento, juntamente com as suas principais características e os seus primeiros testes individuais.

- **Capítulo 4 - Implementação do protótipo:** descreve todo o desenvolvimento do protótipo, tanto da parte hardware como software.

- **Capítulo 5 - Resultados:** apresenta a análise dos resultados conseguidos e as observações relativas aos mesmos.

- **Capítulo 6 - Conclusão:** exprime as principais conclusões relativas ao trabalho desenvolvido e sugestões para um aperfeiçoamento no futuro.

Capítulo 2

Estado da arte

A medicina veterinária dispõe de um vasto campo de atuação cujo objetivo é a melhoria dos cuidados de saúde dos animais. As técnicas clínicas para monitorizar a saúde dos animais foram sendo consideradas insuficientes, uma vez que estas forneciam apenas informações casuais e requeriam um alto investimento quer a nível de recursos, como em termos de tempo e conhecimentos veterinários [33, 34]. Desta forma, com a evolução das tecnologias foram feitos avanços nos mecanismos de monitorização e surgiram vários equipamentos, desde monitores a coleiras, para a medição automatizada dos sinais fisiológicos. Este avanço foi crescendo de forma exponencial, justificado pela elevada margem de segurança que transmite, tanto ao animal como ao clínico [35].

Inicialmente, começaram a ser comercializados monitores para as variáveis cardíaca e respiratórias, durante a cirurgia, ajudando no controlo da anestesia [4]. Mais recentemente, a evolução decorre em torno das tecnologias sem fio. Estas permitem não só a monitorização em contexto clínico, o principal foco, mas também o controlo da saúde animal nos seus lares através de aplicações utilizadas pelos próprios tutores. A incorporação de tecnologias Bluetooth foram as primeiras a surgir, principalmente chips e coleiras GPS, para o controlo da localização animal [5]. No entanto, houve necessidade de evoluir essas tecnologias, criando sistemas que detalhassem o estado do animal, e aí, para além da comunicação Bluetooth surgiu a utilização do Wi-Fi. Este tipo de comunicação permite a interação entre pessoas e máquinas, neste caso específico permite acompanhar, monitorizar, estudar, compreender e atuar de imediato sobre o animal, mesmo à distância [36]. No desenlace desta utilização, surgiu o conceito de Internet of Things (IoT) que permite a interconetividade de dispositivos, ou seja, permite que certos equipamentos, sem características computacionais, obtenham funcionalidades de leitura e comunicação através de sensores e atuadores [37]. Este método inicia uma realidade cada vez mais presente, visto que é um processamento de baixo custo, apresentando ainda diversas vantagens na produtividade e otimização de processos.

Atendendo a todos os parâmetros e características referidas anteriormente, foram então surgindo vários equipamentos, tanto para os tutores como a nível hospitalar. Na tabela 2.1 é possível encontrar informação de alguns sistemas de monitorização existentes no mercado.

Tabela 2.1 - Informações acerca de alguns sistemas de monitorização.

	VARIÁVEIS DE MONITORIZAÇÃO	VANTAGENS	DESVANTAGENS	REFERÊNCIAS
Chip Biothermo 	T Leituras feitas a partir de um scanner	Durabilidade de 25 anos; Fidedigno e compatível com o organismo.	Muitas vezes deixam de funcionar devido a interferências de outras fontes.	[38, 39, 40]
Monitor PM10A 	ECG FR Saturação de oxigênio	É possível ligar à central de monitorização.	Máximo de armazenamento de dados de 720h.	[41]
Monitor Datalysis V7 	T ECG FR Saturação de oxigênio	Medição de sinais para qualquer porte animal; Alertas visuais e sonoros.	Preço elevado.	[42]
Monitor FX4000 	T FR ECG Saturação de oxigênio	Portátil; Pra uso cirúrgico e de internamento.	Apenas para animais de porte pequeno.	[43]
Oxímetro ARSTN 	Saturação de oxigênio FC	Portátil; Medições rápidas; Adequado para uso hospitalar.	Taxa de pulso 25-250 bpm; Animais de pequeno porte.	[44]

Dispositivo SURE

Atividade física
Consumo calórico
Qualidade do sono

À prova de água e de pó;
Muito leve, fácil colocação na coleira;
Possibilidade de partilhar dados com o centro clínico.

Raio de sincronização < 10m;
Capacidade de armazenar dados < 3 semanas.

[45]

Coleira PetPace

T
FR
FC
Atividade física
Calorias consumidas

Envia alertas, em caso de anomalias, simultaneamente para o tutor e para o centro veterinário;
Bateria dura 6 semanas e é recarregável.

Preço diverge entre animais saudáveis e com doenças.

[46, 47]

Coleira Voyce pro

T
FR
FC
Calorias consumidas
Qualidade do sono
Atividade física

Uso hospitalar (monitoramento pré e pós cirúrgico).

Para além de ser necessário a compra da coleira, a sua aplicação também é cobrada.

[48, 49, 50]

Coleira Buddy

T
FC
Quantidade de atividade realizada
Calorias consumidas
Localização

Envia avisos quando é detetada alguma anomalia;
À prova de água;
Bateria dura 14 dias e é recarregável.

Dados são enviados apenas para a aplicação dos tutores não tendo ligação ao centro veterinário.

[51]

Dispensador Petnet

Quantidade de comida
Consumo de calorias

Envio de notificações;
Aviso de comida insuficiente no recipiente.

Necessita de software de smartphone específico.

[52]

Câmara de vigilância PetCube

Quantidade de comida

Vigia do animal com função de visão noturna;
Sistema de som com 2 vias;
Dispensador de snacks.

Ângulo de visão de 138°;
Capacidade do recipiente de 1kg.

[53]

A análise da tabela 2.1 permite concluir que existem vários equipamentos de monitorização disponíveis no mercado desde dispensadores de comida, a monitores para cirurgia e internamento, até coleiras que medem parâmetros para o controlo animal. No entanto, o elevado preço destes sistemas faz com que ainda não sejam muito utilizados, principalmente por parte dos centros hospitalares veterinários. Normalmente, o preço de um monitor para um centro clínico pode variar entre os 1500 e os 5000€, sendo necessário um para cada animal na fase de recobro. Isto repete-se para todos os dispositivos de monitorização apresentados, tornando a sua aquisição extremamente dispendiosa.

A componente de produção animal para alimentação de seres humanos é também bastante importante. Através da alimentação, o bem-estar animal tem repercussões na saúde humana. Assim é crucial ter todos os cuidados com estes, da forma mais confortável, económica e de fácil utilização para os agricultores [54]. Mais recentemente foram desenvolvidos vários protótipos para a monitorização animal nesta área, tendo como base a IoT, uma vez que permite uma instalação fácil e eficaz. A maioria dos sistemas desenvolvidos focam no uso do Arduino, com sensores colocados no animal que com comunicação Wi-Fi, envia os dados dos sinais vitais em tempo real para o tratador. Os principais sinais medidos são temperatura, frequência respiratória, pressão arterial, frequência cardíaca e a ruminação, ou seja, a quantidade de comida que estes consomem. Com base nos dados recolhidos é possível, praticamente de forma automática, ver se o animal está de boa saúde ou não, facilitando a deteção de doenças e também a relação com os médicos veterinários, uma vez que vão conseguir prescrever medicamentos adequados no momento certo, o que pode reduzir o custo de posteriores tratamentos [54, 55, 56]. Para além das vantagens de controlo que este sistema dá ao agricultor, é ainda de salientar o baixo consumo, baixo custo e resposta rápida [55].

Estes sistemas desenvolvidos para os meios rurais são essenciais para o bem-estar do animal. O grande problema é que estes não se encontram, ainda, em meio clínico, principalmente na fase pós-cirúrgica, sendo esta uma necessidade urgente. Tendo em conta o custo de alguns sistemas apresentados na tabela 2.1, a aposta é focar em sistemas como os que têm sido desenvolvidos para o controlo agrícola dos animais, mas automatizados para serviço clínico. Isto irá permitir um melhor cuidado do animal, com mais segurança e conforto.

2.1 - Necessidade hospitalar

Após uma análise da literatura e dos sistemas existentes, foi necessário pesquisar o que os hospitais veterinários realmente procuravam no sentido de responder às suas necessidades. Para tal, com a ajuda de diferentes centros como o i3S (instituto de investigação e inovação em saúde), o hospital veterinário ANIMALcare no Porto e o HVUTAD (Hospital veterinário da Universidade de Trás-os-Montes e Alto Douro) foram realizados inquéritos a vários médicos veterinários (Anexo A). Foi-lhes questionado a importância da monitorização, parâmetros cruciais, falhas ainda existentes nas medidas de controlo e de que forma um sistema destes poderia inovar e melhorar o recobro pós-cirúrgico.

Esta secção resume as principais conclusões retiradas da análise dos inquéritos realizados. A análise geral das respostas levou a uma melhor perceção da necessidade dos centros clínicos, permitindo assim uma escolha mais adequada do sistema dos parâmetros a medir.

O controlo do animal na fase pós-cirúrgica pode ser crucial para o mesmo, devido aos efeitos de fármacos, principalmente a anestesia, como já mencionado anteriormente. Desta forma as

respostas afirmativas às questões “Acha importante a criação de um sistema, onde à distância, seja possível controlar os parâmetros cruciais para a recuperação do animal?” e “Acha importante a sua implementação em meio hospitalar?” foram unânimes. Todos os clínicos sublinharam a importância da criação e implementação de um sistema de monitorização nesta fase, impondo a ideia que permitiria um melhor controle do animal e de uma forma muito mais cómoda para ambos.

Contudo, é necessário avaliar os problemas presentes atualmente nessa fase. Nesse sentido para conseguir melhorá-los ou até mesmo resolvê-los, foi colocada a questão de “Quais considera os maiores problemas na monitorização, na fase de recobro?”.

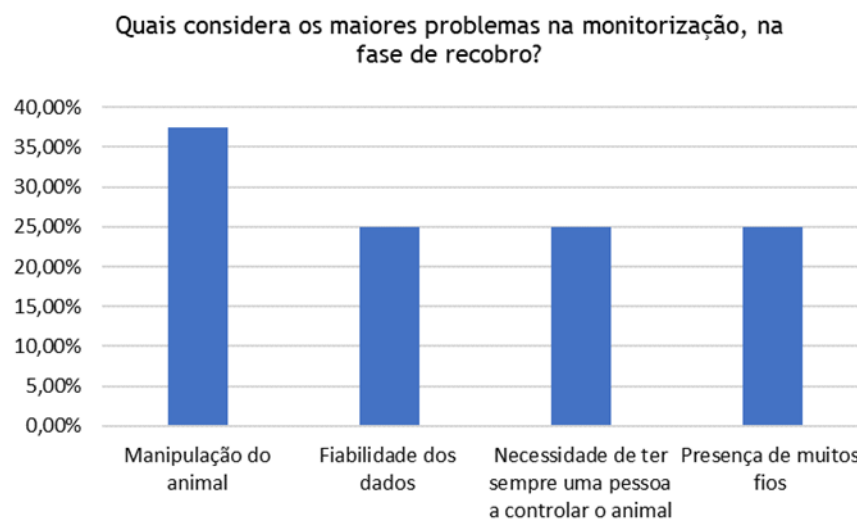


Figura 2.1 - Análise gráfica da resposta à segunda questão do inquérito: “Quais considera os maiores problemas na monitorização, na fase de recobro?”

De um modo geral as respostas foram variando, no entanto, verificaram-se sempre pontos comuns que geram os maiores problemas, estando estes representados na figura 2.1. A manipulação do animal é das maiores preocupações para a equipa clínica, uma vez que, ao mexer com este enquanto está em recuperação pode causar danos e ferimentos irreversíveis. A manipulação altera também as medidas fisiológicas a recolher durante a observação. A presença de muitos fios nos aparelhos de controlo e a necessidade de ter sempre uma pessoa a monitorizar os sinais vitais do animal constitui um incómodo para ambas as partes. Esta questão permite retirar alguns fatores cruciais a melhorar ao criar um sistema de monitorização. No entanto, é preciso aprofundar os parâmetros mais importantes a monitorizar no período pós-cirúrgico. Nesse sentido a pergunta de “Quais considera ser os parâmetros mais importantes a monitorizar?” permite retirar importantes conclusões.

Na fase de recobro, quais os parâmetros mais importantes a monitorizar?

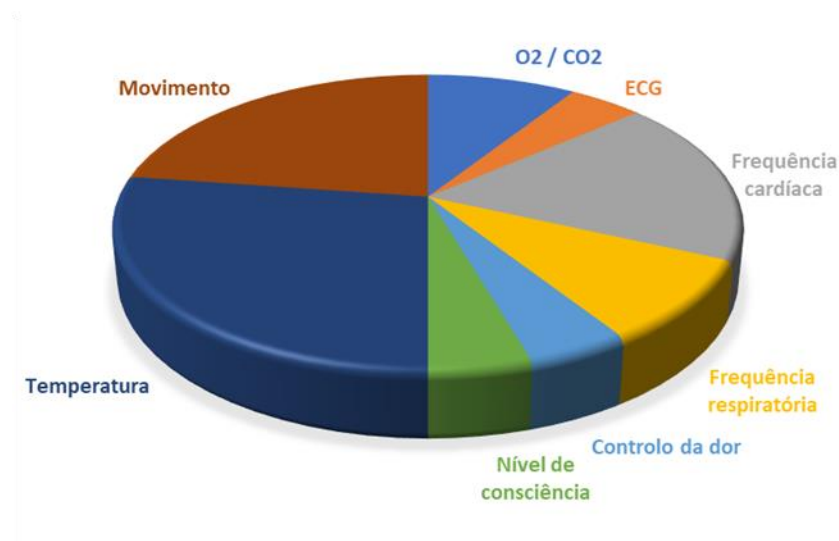


Figura 2.2 - Análise gráfica da resposta à terceira questão do inquérito: “Na fase de recobro, quais os parâmetros mais importantes a monitorizar?”

Com base nas respostas recolhidas, foi possível retirar vários parâmetros importantes a ter em atenção nesta fase de recobro do animal. A temperatura e o movimento são responsáveis por cerca de 50% do estudo efetuado, assumindo-se assim que estes são os parâmetros críticos. De seguida surge a frequência cardíaca e respiratória, restando por último o controlo de oxigénio, da dor, nível de consciência e ECG. Uma curiosidade é que instrumentos são usados para a monitorização e que limitações apresentam.

Tabela 2.2 - Respostas recolhidas sobre os equipamentos existentes para a monitorização e as limitações que estes apresentam.

EQUIPAMENTOS	LIMITAÇÕES
Pulso oxímetro	Muitos fios Higiene do equipamento Invasivos Manutenção complicada das sondas
Eletrocardiograma	
Termómetro	
Estetoscópio	
Tubos de Saturação	

Dos equipamentos referidos é possível associar limitações comuns a todos eles, como é possível aferir na tabela 2.2. Assim, foi ainda recolhido junto dos médicos as vantagens e mais valias que um sistema de monitorização à distância iria trazer para estes centros clínicos.

Tabela 2.3 - Vantagens do sistema recolhidas pela opinião de clínicos.

VANTAGENS E MAIS VALIAS DO SISTEMA
• Permitiria a avaliação de vários animais ao mesmo tempo • Possível monitorização mesmo sem a presença de pessoal médico • Menos manipulação do animal • Processo independente • Qualquer clínico ia conseguir monitorizar o animal

Por último, analisando todas as vantagens que ofereceria e o quão útil seria implementar um sistema destes, foi necessário definir em que animais iria ser inicialmente aplicado. Assim, com a ajuda dos clínicos foi possível selecionar as espécies mais atrativas - o gato e o cão - para a aplicação do sistema e recolher ainda sugestões para a sua melhoria.

**Figura 2.3** - Análise gráfica da resposta à sétima questão do inquérito: “Implementando um sistema destes, quais as espécies em que prevê maior utilização?”.**Tabela 2.4** - Sugestões enunciadas pelos clínicos como resposta à última questão do inquérito: “Caso queira deixar alguma sugestão ou informação que lhe tenha surgido ao longo deste questionário, no sentido de melhorar a ideia ou até mesmo o dispositivo, coloque aqui”.

SUGESTÕES
• Ter o controlo de comida e bebida ingerida pelo animal • Controlo da medicação • Controlar o peso do animal • O mínimo de fios possíveis • Incorporação em coleiras ou jaula, sem interferir com o animal • Ligar automaticamente às fichas clínicas • Não obrigar o animal a estar limitado ao movimento

Finalizado o inquérito, e analisando as respostas foi possível retirar informações e características para a criação do sistema de monitorização a desenvolver. É realçada a importância de haver sistemas de monitorização à distância para controlo do animal, principalmente se a sua utilização for em cães ou em gatos, tentando diminuir a necessidade de ter sempre a presença de médicos e enfermeiros a monitorizar, a quantidade de fios e principalmente a manipulação do animal. A temperatura e o movimento são, sem dúvida, dois dos parâmetros cruciais a monitorizar na fase de recobro, havendo já vários equipamentos para a monitorização, mas que apresentam limitações, como a quantidade de fios e a sua higiene. A utilização de um sistema destes permitiria avaliar vários animais ao mesmo tempo sem ser necessário pessoal médico, tornando-se um processo independente.

Com a ajuda prestada pelo corpo clínico, foi possível responder às necessidades atuais, apresentando um sistema apto e mais vantajoso do que os dispositivos existentes.

Capítulo 3

Protótipo a desenvolver

A escolha do protótipo a desenvolver teve como base, não só alguns mecanismos de sistemas já funcionais, mas também, e principalmente, as características que os clínicos enunciaram como importantes. O requisito principal baseou-se no fator - comodidade do animal -, ou seja, conseguir efetuar a monitorização via Wi-Fi, garantindo a diminuição do número de fios do sistema e evitando que lhes sejam colocados os sensores diretamente, de forma a não causar qualquer distúrbio no seu movimento e bem-estar. A partir daí, a finalidade foi conseguir medir os parâmetros mais importantes na sua recuperação, da forma mais económica possível, uma vez que muitos dos sistemas já desenvolvidos não se encontram em funcionamento hospitalar devido ao investimento necessário.

Desta forma, o protótipo proposto consiste num sistema de monitorização sem fios, para animais de companhia, cães e gatos, especificamente para a fase de recobro pós-cirúrgico. Este deve permitir a medição de parâmetros como a temperatura corporal, o movimento, a quantidade de comida e bebida ingerida e ainda a possibilidade de vigília do animal.

A solução tem como base o Arduino UNO, onde estão ligados todos os sensores, juntamente com um módulo wireless que comunica com a base de dados e a aplicação desenvolvida. Toda a seleção do material, foi feita minuciosamente de forma a garantir a melhor relação preço-qualidade para a finalidade desejada. Para a medição da temperatura animal, será usado um sensor térmico de infravermelhos. Inicialmente a solução passava por colocar uma câmara térmica (Thermal Camera Breakout), no entanto, devido ao COVID, o produto esgotou no mercado e foi necessária outra abordagem, que levasse à mesma finalidade. No caso do movimento, serão colocadas células de carga na base inferior da jaula do animal, e a partir daí será feita a avaliação do seu movimento. Esta monitorização do movimento será acompanhada com a utilização de uma câmara que permitirá não só ao clínico ver o seu movimento através de imagens recebidas, mas também do lado dos tutores “visitarem” o seu animal online, diminuindo a necessidade destes se deslocarem aos centros clínicos. De forma a obter a quantidade de bebida e comida ingerida pelo animal, serão usadas, novamente, células de carga, para obter o valor da quantidade de comida e bebida presente nos respetivos recipientes, obtendo assim o balanço daquilo que o animal ingeriu ao longo de toda a sua recuperação. Por fim, para o seguimento e

controle do ambiente em que o animal se encontra é utilizado um sensor de temperatura e humidade para lhe fornecer as condições necessárias. É de salientar que nenhum dos sensores utilizados será colocado no animal, mas sim na jaula, proporcionando-lhe assim um maior conforto. Na figura 3.1 é possível, através do desenho de arquitetura, ter uma perceção mais real do sistema a desenvolver. Como é possível analisar, destacou-se o uso de dois módulos Wi-Fi, opção tomada para um melhor e mais rápido processamento e envio dos dados recolhidos, sem ocupar a totalidade da memória dos microcontroladores.

A aplicação online, disponível para computador, recebe os dados dos sensores via Wi-Fi, armazenando-os na base de dados, permitindo visualizar, posteriormente, os parâmetros recolhidos. Além desta monitorização contínua, a ideia é o sistema alertar o médico em caso de anomalia, através de avisos sonoros, aumentando a segurança do animal e do corpo clínico nesta fase de recobro.

As secções a seguir descrevem os componentes utilizados e as suas comunicações individuais para os primeiros testes de funcionamento, antes de juntar todos num só sistema.

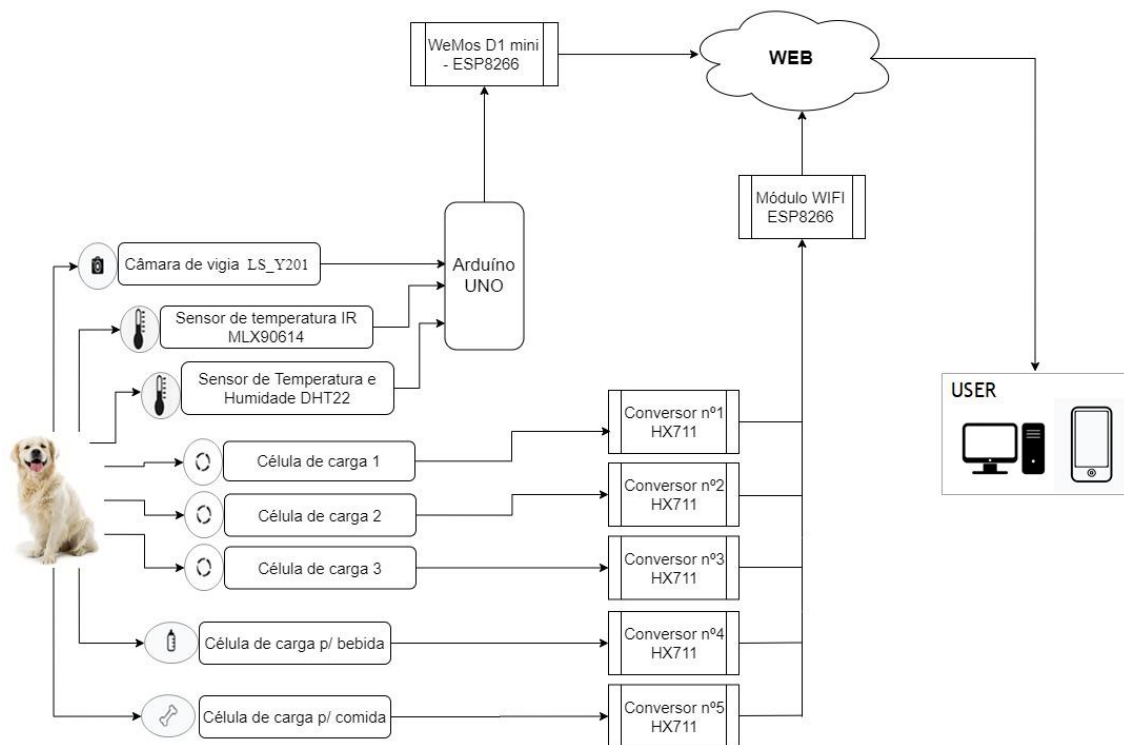


Figura 3.1 - Desenho de arquitetura.

3.1 - Arduíno UNO

O Arduíno UNO (figura 3.2) é uma placa de desenvolvimento baseada no microcontrolador ATmega328P, com 2 KB de SRAM (memória estática de acesso aleatório) e 32 KB de memória flash [57]. A primeira, para armazenar cada bit, usa memória volátil, sendo os dados perdidos quando a energia é removida [58]. A segunda, é uma memória não volátil, podendo ser eliminada e reprogramada eletricamente [58].

A placa contém 14 pinos de entrada/saída digitais e analógicas que permite a conexão com outras placas de expansão ou outros circuitos [57]. Inicialmente, cada placa precisa de uma fonte de alimentação, no caso do arduino este pode ser alimentado por um cabo USB (figura 3.2 (1)) ou por uma alimentação com um conector cilíndrico (figura 3.2 (2)), sendo a voltagem recomendada entre 6 e 12 volts (V) [59]. Existem vários pinos de Ground (GND) (figura 3.2 (3)) que podem ser usados. Para fornecer energia aos componentes usados existem pinos de 5V (figura 3.2 (4)) e de 3,3V (figura 3.2 (5)) [59]. A vertente analógica do arduino está identificada por “Analog In”, onde estão inseridos os pinos de A0 a A5 (figura 3.2 (6)), estes permitem a leitura de sensores analógicos e a conversão num valor digital [59]. Do lado oposto estão presentes os pinos digitais, do 0 ao 13 (figura 3.2 (7)), usados como entrada ou como saída digital [59]. Dentro destes é possível, ainda, diferenciar alguns deles, os que têm um “~” próximo dos números (figura 3.2 (8)), são diferenciados por permitirem uma modulação por largura de pulso (PWM), ou seja, são capazes de simular uma saída analógica [59]. Como referência analógica, se for o pretendido, é possível definir uma tensão externa entre 0 e 5V através do AREF (figura 3.2 (9)) [59]. Existe ainda um botão reset (figura 3.2 (10)), que permite reiniciar qualquer código carregado no Arduino, o LED de energia (figura 3.2 (11)), que acende assim que o Arduino é conectado à fonte de alimentação e os LEDs de comunicação TX (transmitir) e RX (receber) (figura 3.2 (12)), que dão a indicação visual se estamos a receber ou transmitir dados [59]. Por fim, temos o microcontrolador integrado (figura 3.2 (13)), ATmega, e o regulador de tensão (figura 3.2 (14)), que permite aceitar alimentações externas até 12V [59].

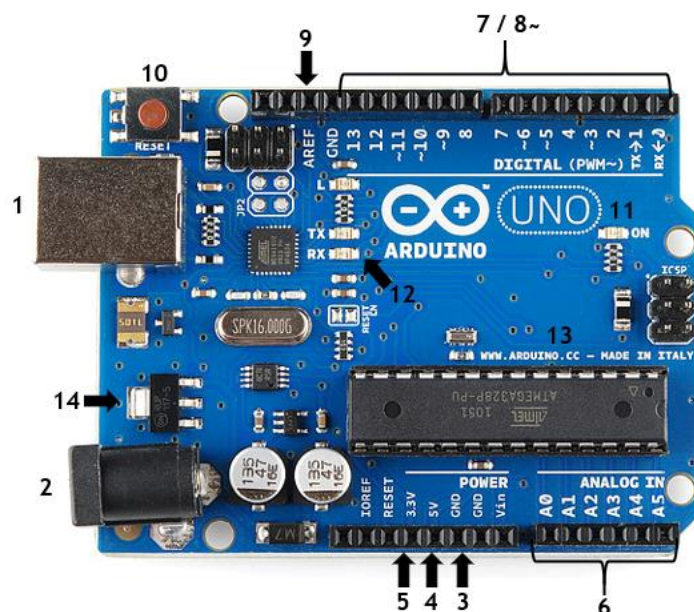


Figura 3.2 - Arduino UNO [59].

O facto do Arduino ser uma tecnologia open source, com hardware económico e software gratuito, tem uma enorme documentação disponível para os mais diversos projetos e uma capacidade elevada para conexão com os mais diversos sensores, faz com que seja a escolha ideal para o protótipo pretendido. A sua função passa por receber e processar os dados dos sensores, do sistema, a ele ligados.

3.1.1 - Arduíno IDE

O Arduíno IDE (Integrated Development Environment) é o software utilizado para a programação do Arduíno UNO. Este permite a programação de diferentes tipos de placas desde que sejam feitos os ajustes e inseridas as bibliotecas e definições necessárias. A extensão do arquivo é *.ino* e a estrutura do código é dividida em `setup()` e `loop()` (figura 3.3). A primeira, a função `setup()` é ativada quando o código inicia, no entanto, esta é executada apenas uma vez, após cada inicialização, assim como o modo de funcionamento dos pinos de (I/O) e os valores iniciais definidos. A função `loop()`, tal como o nome indica, está continuamente em loop, permitindo assim que o programa responda às condições instantâneas.

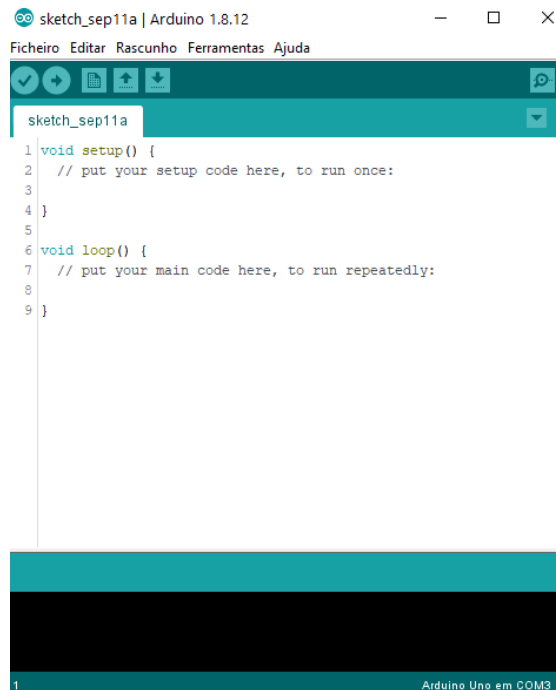


Figura 3.3 - Layout do Arduíno IDE.

3.2 - Placa de desenvolvimento ESP8266 NodeMCU

O módulo ESP8266 NodeMCU (figura 3.4) é uma placa de desenvolvimento com o chip ESP8266-12E integrado (figura 3.4 (1)). É um dos 12 modelos do ESP8266, um módulo com um microcontrolador integrado que contém uma antena Wi-Fi [60]. Este possui uma memória flash de 4 Mb, uma entrada micro USB para alimentação e programação (figura 3.4 (2)), um conversor USB serial integrado (figura 3.4 (3)), um regulador de tensão e Wi-Fi nativo padrão 802.11b/g/n (protocolo de comunicação utilizado) [60], [61]. Contém apenas 1 pino analógico com uma resolução de 10 bits (figura 3.4 (4)) e 11 pinos digitais, a maioria (10) com função PWM (figura 3.4 (5)) [60]. Junto da entrada micro USB tem o botão RST (reset) (figura 3.4(6)) e o botão de FLASH (figura 3.4(7)) que permite a gravação do programa no ESP-12 [61]. A sua programação pode ser feita utilizando a IDE do arduíno, que é facilitada pelo uso de diversas bibliotecas já existentes [60], [61].

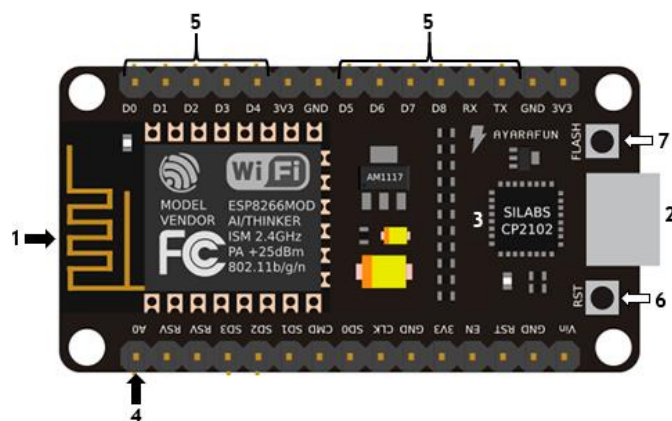


Figura 3.4 - NodeMCU ESP8266-ESP12.

3.3 - WeMos D1 mini NodeMcu - ESP8266

A WeMos D1 Mini NodeMcu - ESP8266 (figura 3.5) é uma placa de desenvolvimento com o módulo ESP8266 integrado, de 4 Mb de memória flash Wi-Fi [60]. Apresenta uma entrada micro-USB (figura 3.5(1)) e duas saídas de alimentação para componentes externos, uma de 3.3V (figura 3.5 (2)) e outra de 5V (figura 3.5 (3)). Ambos os pinos conseguem fornecer até 500 mA, sendo que, o pino de 5V é capaz de alimentar toda a placa com uma tensão de alimentação que varia entre os 4V e os 6V [62]. Sendo este, uma das versões mais reduzidas do NodeMcu, faz com que tenha apenas 8 entradas digitais (figura 3.5 (4)), no entanto, todas elas são igualmente capazes de produzir sinal PWM. Quanto às entradas analógicas, apresenta apenas uma (figura 3.5 (5)). Os restantes pinos são de comunicação, o RX (figura 3.5 (6)) e o TX (figura 3.5 (7)), o GND (figura 3.5 (8)) e o botão RST, reset (figura 3.5 (9)).

A sua utilização é bastante versátil, pois este apresenta todos os pinos de comunicação, UART, SPI e I2C [62]. Quanto à programação, como a maioria dos ESP8266, pode ser feita no arduino IDE.

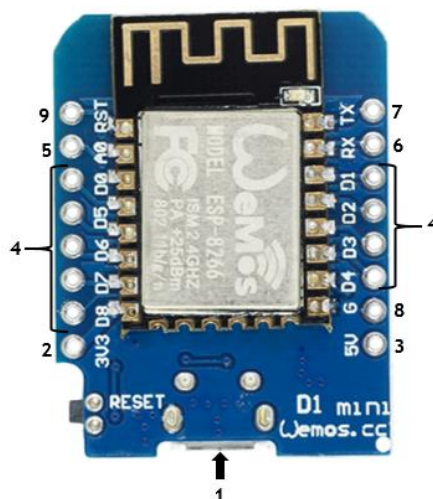


Figura 3.5 - WeMos D1 mini NodeMcu - ESP8266.

3.4 - Sensor de temperatura e humidade DHT22

O sensor DHT22 / AM2302 (figura 3.6), integra um sensor de humidade e um sensor de temperatura num só módulo. Utiliza um sensor capacitivo de humidade e um termistor com uma elevada precisão, que conectados ao seu microcontrolador de 8 bits, produzem um sinal digital [63]. A sua gama de medição está entre -40°C a 80°C no caso da temperatura e 0 a 100% na humidade, com uma margem de erro de 0.5°C para o primeiro e de 2% para o segundo caso, respetivamente [63].

Quanto ao número de pinos, este contém apenas 4. O pino de alimentação (VDD) (figura 3.6 (1)), o qual suporta uma tensão entre 3.3V e 5.5V, sendo os 5V o valor recomendado, o SDA (figura 3.6 (2)) que é a porta serie, o NC (figura 3.6 (3)) é um pino vazio, não é utilizado, e por fim o GND (figura 3.6 (4)) [63].



Figura 3.6 - Sensor DHT22.

3.4.1 - Programação no arduíno IDE

A ligação do sensor DHT22 ao arduíno UNO é básica, uma vez que este apresenta apenas 3 pinos (que fazem ligações) e não é necessária a utilização de componentes externos para a sua funcionalidade. Assim, é apenas necessário ligar o seu VDD aos 5V do arduíno, o SDA a uma porta digital, neste caso o pino D7, e por fim ligar o GND ao GND do aduíno como mostra a figura 3.7

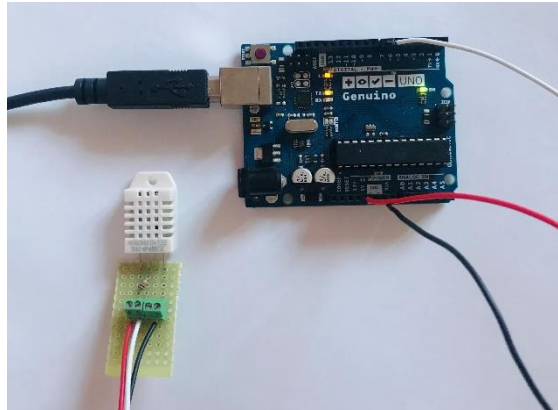


Figura 3.7 - Conexão do sensor DHT22 ao arduino UNO.

A sua programação é feita no arduino IDE. Tem a facilidade de já existir uma biblioteca que pode ser instalada no IDE e que inclui o código deste sensor (figura 3.8).

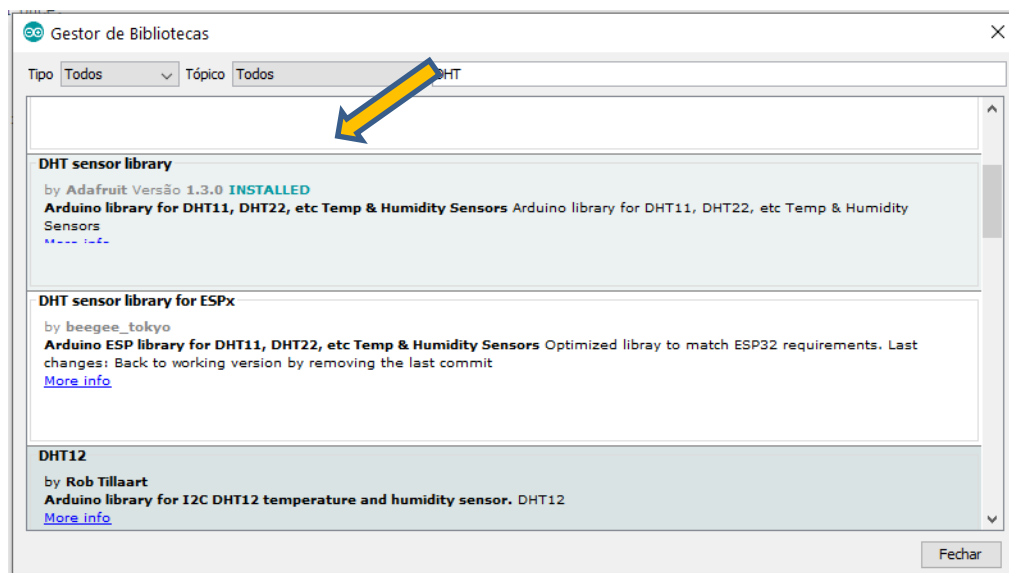


Figura 3.8 - Instalação da biblioteca do DHT22.

O código começa, depois de incluir a biblioteca referente, por definir o pino digital do Arduino, o modelo do sensor de temperatura e humidade escolhido e colocar a função que revela os valores pretendidos (figura 3.9).

```
1 #include <DHT.h>
2 #include <DHT_U.h>
3 #include <DHT.h>; //Inclusão da biblioteca
4
5 #define DHTPIN 7 //Pino digital utilizado
6 #define DHTTYPE DHT22 //Definição do modelo do sensor utilizado (DHT22)
7
```

```
8  DHT dht(DHTPIN, DHTTYPE); //Pôr os parâmetros na função
9
10 void setup(){
11   Serial.begin(9600); //Inicia a serial
12   dht.begin(); //Inicia a função
13   delay(2000); //Intervalo de 2 segundos antes de começar
14 }
15 void loop(){
16   Serial.print("Humidade: "); //Texto que quero imprimir no monitor
17   Serial.print(dht.readHumidity()); //Imprime o valor de humidade medido
18   Serial.print("%"); //Imprime o sinal de %
19   Serial.print(" / Temperatura: "); //Texto que quero no monitor
20   Serial.print(dht.readTemperature(), 0); //Imprime o valor de temperatura medido
21   Serial.print("°C"); //texto
22   delay(2000); //Intervalo de 2 segundos - neste caso não pode ser menos porque
23     este é o tempo de resposta do sensor
24 }
```

Figura 3.9 - Código de programação para o sensor DHT22.

3.5 - Sensor de temperatura IR MLX90614

O MLX90614 (figura 3.10) é um termómetro infravermelho que permite fazer medições de temperatura sem contacto com o objeto. Contém um amplificador de baixo ruído, um ADC (conversor analógico-digital) de 17 bits e uma unidade DSP (processador de sinal digital). O termómetro sai de fábrica calibrado com uma saída digital PWM e SMBus (System Management Bus), sendo que esta última lhe dá um acesso total à temperatura medida nas faixas completas. No caso do PWM, este é de 10 bits e é configurado para, continuamente, transmitir a temperatura medida numa gama de -20 a 120°C [64].

O valor obtido do sensor é a temperatura média de todos os objetos que se encontram no seu campo de visão. Este permite fazer a medição da temperatura ambiente, tendo uma gama entre -40°C e 125°C, e a medição de um corpo/objeto em concreto, com valores compreendidos entre -70°C e 380°C, apresentando em ambos os casos uma precisão de $\pm 0.5^\circ\text{C}$ [64].

A comunicação com o microcontrolador é feita por I2C, através dos pinos SDA e SCL. Os outros 2 pinos, sendo que o MLX90614 apresenta apenas 4, são o VCC, onde a alimentação é de 5V, e o GND (figura 3.10) [64].



Figura 3.10 - Sensor de temperatura IR MLX90614.

3.5.1 - Programação no arduino IDE

A programação do MLX90614 pode ser feita no arduino IDE. O primeiro passo é fazer a conexão do mesmo ao arduino UNO. Este passo é relativamente simples devido ao baixo número de pinos que o sensor apresenta. O pino GND do MLX90614 vai ligar ao GND do arduino, o VCC aos 5V e por fim o SDA e o SCL aos pinos A4 e A5, respetivamente (figura 3.11).

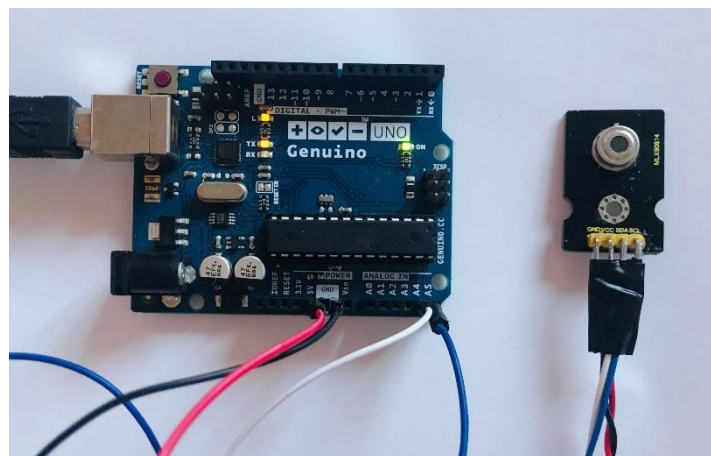


Figura 3.11 - Conexão do MLX90614 ao arduino UNO.

A parte do código para o seu funcionamento começa pela inclusão das bibliotecas do sensor, a “Wire.h” e a “Adafruit_MLX90614.h”. Estas são instaladas através do gestor de bibliotecas (opção selecionada em “ferramentas” do menu do IDE), como demonstrado na figura 3.8, e já incluem todas as funções necessárias para que seja possível criar uma rotina de forma a obter os valores pretendidos. Esta passa por imprimir o valor medido junto de um simples texto que nos indica a temperatura que está a medir, como é possível ver na figura 3.12.

```

1  #include <Wire.h>
2  #include <Adafruit_MLX90614.h>
3  Adafruit_MLX90614 mlx = Adafruit_MLX90614();
4
5  void setup() {
6    Serial.begin(9600);
7    Serial.println("Adafruit MLX90614 test");
8    mlx.begin();
9  }
10 void loop() {
11   Serial.print("Ambient = "); Serial.print(mlx.readAmbientTempC());
12   Serial.print(" *C\tObject = "); Serial.print(mlx.readObjectTempC());
13   Serial.println(" *C");
14   /* Serial.println(); */
15   delay(500);
16 }

```

Figura 3.12 - Código de programação para o sensor MLX90614.

3.6 - Câmera LS_Y201

LS_Y201 (figura 3.13) é um módulo de câmera (32mm x 32mm) capaz de capturar imagens a cores, em JPEG, usando a comunicação série [65]. Utiliza uma interface UART (Universal Asynchronous Receiver/Trasmitter), o que lhe permite comunicar a uma maior distância e que a sua sincronização seja feita por software. O tamanho máximo de imagem que o módulo permite é de 640x480 pixels, o que, juntamente com a sensibilidade que apresenta à luz infravermelha (alterando a reprodução de cores), faz com que a sua resolução não seja extraordinariamente elevada, mas o suficiente para ser uma perfeita câmera de vigilância [66].

Para a conexão da câmera com o microcontrolador, neste caso o arduino, é necessário usar 4 pinos, o de 5V para a alimentação, o GND para aterrar o circuito, o RX para o recebimento de dados e por último o TX para a transferência de dados [66]. Estes são visíveis no próprio módulo, como pode ser observado na figura 3.13.

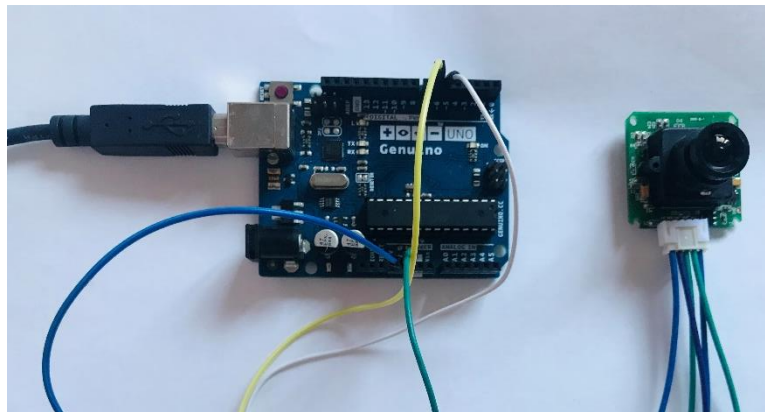


Figura 3.13 - Câmara LS_Y201 [66].

O armazenamento dos dados recolhidos pela câmara, normalmente, é feito externamente, por exemplo, num microSD [66]. No entanto, no protótipo a desenvolver o armazenamento será feito diretamente na aplicação desenvolvida (a sua conexão é feita ao Arduino, mas os dados serão enviados por Wi-Fi), daí a câmara ser usada igualmente como vigia, mas com imagens, atualizadas em 2/3 segundos.

3.6.1 - Programação no arduino IDE

O primeiro passo é fazer as ligações da câmara ao arduino UNO. O GND liga ao GND do arduino, o 5V ao respetivo 5V e por fim o RX e o TX ligam aos pinos digitais 5 e 6, por esta ordem (figura 3.14).

**Figura 3.14** - Conexão da câmara LS_Y201 ao arduino UNO.

A programação é feita no Arduino IDE iniciando com a inserção das bibliotecas necessárias para o software. Neste caso, é a “SoftwareSerial.h” e a “LSY201.h” que podem ser instaladas no gestor de bibliotecas (figura 3.8) ou, nas versões do IDE onde estas ainda não estiverem presentes, o seu download pode ser feito no GitHub (plataforma OpenSource de códigos).

O seu código começa pela inclusão das bibliotecas (regra geral de todos os códigos) indicando-se seguidamente os canais que estão a ser usados para fazer a comunicação da câmara com o microcontrolador Arduino UNO. Por fim, é criada a rotina, neste caso o envio da imagem em hexadecimal que posteriormente, neste caso em específico, será guardado em ficheiro para a base de dados. É possível ver a rotina criada na figura 3.15.

```
1 #include <SoftwareSerial.h>
2 #include <LSY201.h>
3
4 /* assuming the TX and RX pins on the camera are attached to pins 5 and 6 of
5  * the arduino. */
6 SoftwareSerial camera_serial(5, 6);
7
8 LSY201 camera;
```

```
9  uint8_t buf[32];
10 uint8_t stop_flag=0;
11 void setup()
12 {
13   camera.setSerial(camera_serial);
14   camera_serial.begin(38400);
15   camera.setImageSize(0x11);
16   camera.reset();
17   camera.setCompressionRatio(0x80);
18   Serial.begin(115200);
19 }
20
21 void loop()
22 {
23   Serial.println("Taking picture...");
24   camera.takePicture();
25
26   Serial.println("Bytes:");
27   Serial.println(camera.readJpegFileSize());
28   uint16_t offset = 0;
29   while (camera.readJpegFileContent(offset, buf, sizeof(buf)))
30   {
31     for (int i = 0; i < sizeof(buf); i++){
32       if ((buf[i] == 0xFF) && (stop_flag == 0))
33         stop_flag=1;
34       else if ((buf[i] == 0xD9) && (stop_flag == 1))
35         stop_flag=2;
36       else
37         stop_flag=0;
38       if (buf[i]<16){
39         Serial.print("0");
40         Serial.print(buf[i], HEX);
41       }
42       else
43         Serial.print(buf[i], HEX);
44       Serial.print(" ");
45       if (stop_flag==2)
46         break;
47     }
48     Serial.print("");
49     offset += sizeof(buf);
50     if (stop_flag==2)
51       break;
52   }
53 }
54
```

```

55 Serial.println("Done.");
56
57 camera.stopTakingPictures();
58 delay(100);
59 }

```

Figura 3.15 - Código de programação para a câmara LS-Y201.

3.7 - Células de carga

A célula de carga consiste num transdutor que traduz a força (pressão) num sinal elétrico, sendo este sinal diretamente proporcional à força aplicada [67]. Existem vários tipos de células e de várias faixas, desde hidráulicas, pneumáticas, de extensómetro, no entanto a sua escolha baseou-se no ambiente em que estas vão ser utilizadas.

Neste protótipo específico, as células que mais se adequam são as células de carga de extensómetro, revelam o valor da força através da deformação. Apresentam uma forma em Z, com quatro extensómetros numa configuração de ponte de Wheatstone, facilitando a medição das mais pequenas variações [67, 68]. Quando a força é aplicada sobre a célula, o extensómetro começa a deformar, na mesma proporção em que o peso aumenta, e é essa mesma deformação que vai indicar através de um sinal elétrico qual o peso que está sobre ela (figura 3.16).

Como referido anteriormente estas serão usadas para o controle do movimento do animal e para a monitorização da quantidade de comida e bebida ingerida por este. Assim sendo, para a medição do movimento serão usadas três células de 5kg (o suficiente, uma vez que o protótipo será desenvolvido para animais de pequeno porte) e relativamente à comida e bebida serão usadas células de 500g, colocadas no comedouro e bebedouro.

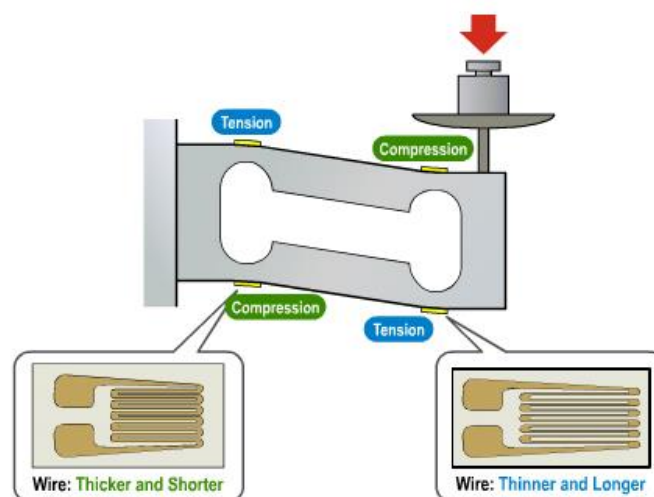


Figura 3.16 - Diagrama da célula de carga [67].

Os sinais elétricos gerados pela célula de carga são da ordem de milivolts (mV) e para os amplificar, normalmente, é usado o sensor HX711 (figura 3.17). Este contém o chip HX711, que é um ADC (conversor analógico-digital) de alta precisão, e tem entradas para alimentar até

duas células de carga ao mesmo tempo. Do seu lado esquerdo ele apresenta os canais E+ e E- para alimentação da célula e os canais A+, A-, B+ e B- para receber o sinal da célula [68]. Do lado oposto tem os canais para comunicar com o microcontrolador (neste caso o Arduino), o VCC, o GND e dois canais analógicos, o DT e o SCK como é possível observar na figura 3.17. Assim, o módulo amplifica a baixa saída elétrica das células de carga e esse sinal é convertido em digital e comunicado ao Arduino.

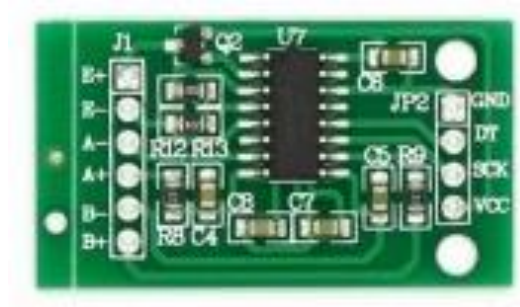


Figura 3.17 - Amplificador HX711.

3.7.1 - Primeira calibração

O primeiro passo é fazer todas as ligações entre a célula de carga e o HX711, e entre este e o Arduino como está indicado na figura 3.18.

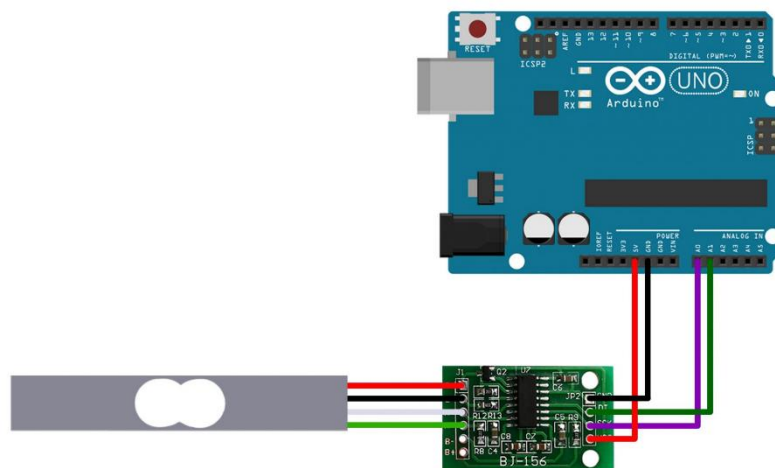


Figura 3.18 - Esquema de ligação entre a célula de carga, o amplificador HX711 e o Arduino UNO [68].

Posto isto, a sua programação é feita no Arduino IDE. Para tal, é necessário instalar a biblioteca do HX711, de seguida definir os pinos do Arduino usados e criar a rotina como é possível ver na figura 3.19.

```
1  #include <HX711.h>           //Incluir biblioteca
2
3  #define pinDT 2              //Definir pinos
4  #define pinSCK 3
5
6  HX711 scale;
7
8  float medida = 0;           //declarar variáveis
9
10 void setup() {
11     Serial.begin(57600);
12
13     scale.begin(pinDT, pinSCK); //Configurar pinos da balança
14     scale.set_scale(50100);     //Calcular valor da escala
15     scale.set_gain(32);
16     delay(2000);
17     scale.tare();
18
19     Serial.println("Balança Zerada");
20 }
21
22 void loop() {
23
24     medida = scale.get_units(5);
25     Serial.println(medida, 3);
26
27     scale.power_down(); //Desliga o sensor
28     delay(500); //espera 5 segundos
29     scale.power_up(); //liga o sensor
30 }
```

Figura 3.19 - Código de programação para a célula de carga.

Capítulo 4

Implementação do protótipo

Este capítulo descreve todo o desenvolvimento do protótipo, contendo a parte de hardware e software. A primeira, consiste na junção de todos os componentes utilizados, a sua montagem na jaula e as suas conexões com os diferentes microcontroladores. Relativamente ao software, inclui o código de programação em IDE com o envio das suas informações para um site, a criação da base de dados, ou seja, o código e a sua funcionalidade, e por fim o desenvolvimento do site criado e as suas funções.

4.1 - Montagem da jaula

Ao longo dos capítulos anteriores foi evidenciado o facto dos sensores utilizados não estarem diretamente em contacto com o animal, mas sim na jaula de forma a não retirar o seu conforto. De maneira a facilitar o processo, construiu-se uma jaula em madeira MDF hidrófugo (escolhido por apresentar fibras de madeira aglutinada com resinas que faz com que esta mantenha as suas características físico-mecânicas mesmo em ambiente húmido), com as medidas 580x580x580 mm, para que fosse o mais parecida possível a uma jaula do hospital veterinário (figura 4.1).

Posto isto, a segunda fase foi colocar todos os sensores na jaula. Foram colocadas as 3 células de carga no chão da mesma (figura 4.1 (A)), de modo a definir o plano da base em contacto com o chão. Esta solução permite determinar o movimento do animal ao longo do seu despertar. É de salientar, que sob estas células assenta um “chão falso”, ligado à jaula apenas pelas células de carga, onde o animal vai ser colocado. Por baixo do seu comedouro e bebedouro foram, também, colocadas células de carga (figura 4.1(B)) para que haja o controlo do que ele come e bebe durante o internamento. Por último, o sensor de temperatura e humidade DHT22, para controlo das condições ambientais, o sensor de temperatura por infravermelhos MLX90614, para controlo da temperatura animal, e a câmara LS-Y201, para a sua vigia, foram presos à parede lateral (figura 4.1 (C1, C2, C3) respetivamente). Sendo que o sensor MLX90614 ficou colocado na direção do comedouro e bebedouro para que fosse possível medir a temperatura do animal quando este estivesse nessa mesma zona.

Como é possível analisar na figura 4.1, o que se vê dos sensores é uma margem relativamente pequena e os seus fios circulam pela parte inferior e lateral da jaula (figura 4.1 (D)), não causando qualquer interferência com o bem-estar animal.

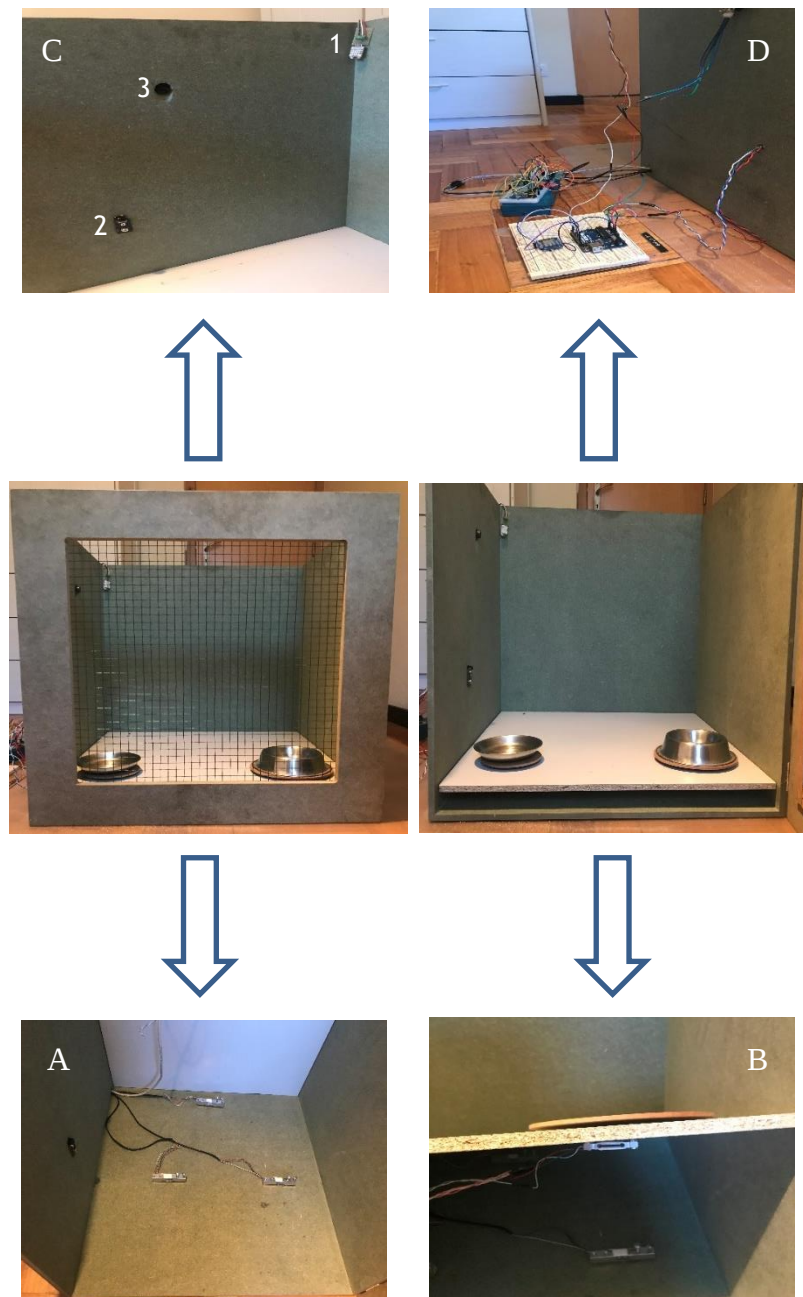


Figura 4.1 - Jaula construída com os sensores colocados.

Como referido no capítulo anterior, para que o sistema corresse à velocidade desejada e sem ocupar muito da memória dos microcontroladores, as comunicações foram divididas em duas partes, ou seja, a decisão foi comunicar as células de carga com o módulo wireless NodeMCU ESP8266-E12 e a câmara LS-Y201, o MLX90614 e o DHT22 foram ligados ao Arduino UNO que faz a leitura dos sensores e envia através do WeMos D1 mini ESP8266.

Nos Anexos B1 e B2 é possível observar o desenho esquemático destas duas comunicações, respetivamente

4.2 - Programação em arduino IDE

Ao longo do capítulo 3, foi mencionado que todos os sensores e microcontroladores utilizados podiam ser programados em arduino IDE. A linguagem utilizada é C++ e a estrutura do seu código é dividida em `setup()` e `loop()`, sendo a primeira executada apenas uma vez quando o código inicia e a segunda está continuamente a ocorrer em loop, tal como o nome indica. Referente, ainda, a este mesmo capítulo, foram indicados os principais passos para começar um código, desde a inclusão das bibliotecas específicas de cada sensor até à criação da rotina. Foram mostrados os códigos utilizados para testar os sensores individualmente, no entanto, agora que estão todos montados na jaula e a trabalhar em conjunto, é necessário fazer um código geral onde sejam chamados os sensores sequencialmente.

Para um melhor funcionamento do sistema, e não sobrecarregar o microcontrolador Arduino e o módulo wireless, os sensores foram divididos pelos dois módulos Wi-Fi. Todas as células de carga ligam diretamente ao NodeMcu ESP8266-12E enquanto todos os outros sensores estão ligados ao arduino e este, posteriormente ao WeMos D1 mini, tal como referido anteriormente. Ao fazer estas ligações, é possível enviar os valores resolvidos pelos sensores, por Wi-Fi para o computador, tornando assim o sistema mais eficiente. É também nesta fase que é feita a conexão dos dados recolhidos com a base de dados, ou seja, os valores que estão a ser constantemente medidos estão a ser enviados via Wi-Fi para o computador e a ser guardados numa base de dados. Esta, por sua vez, irá comunicar esses valores ao site desenvolvido, onde serão analisados graficamente.

4.2.1 - Células de carga - NodeMcu ESP8266-E12

No Anexo B1 é possível verificar as ligações dos sensores ao NodeMCU EPS8266-E12. Os sinais das células de carga entram pelos diferentes canais digitais que o ESP8266 apresenta, os quais têm de ser identificados no código. Inicialmente, é necessário incluir as bibliotecas que são precisas para a utilização dos sensores e das comunicações utilizadas (tabela 4.1). Neste caso é a comunicação por Wi-Fi (do ESP8266 para o computador) e a comunicação HTTP (protocolo de transferência de hipertexto) que permite a troca de dados por web, ou seja, é o que vai permitir que o envio dos valores seja feito para a base de dados [69]. O protocolo utilizado para esta é o TCP/IP (protocolo de controlo de transmissão) que verifica se os dados são enviados sem erros e na sequência correta [69].

Tabela 4.1 - Bibliotecas necessárias para a conexão da célula de carga com o NodeMCU ESP8266-E12.

Biblioteca	Utilidade
ESP8266WiFi.h	Programação do módulo
ESP8266WiFiMulti.h	Possibilidade de registar diversas redes Wi-Fi
String.h	Manipular strings
ESP8266HTTPClient.h	Comunicação HTTP
HX711_ADC.h	Programação do módulo
Stdlib.h	Controlo de processos, conversões e memória

Posto isto, o código é organizado por várias etapas:

- Introdução das bibliotecas;
- Configurações Wi-Fi;
- Definição de funções utilizadas;
- Definição das células de carga;
- Calibração das células de carga;
- **void setup:**
 - Setup dos pinos que detetam as células de carga;
 - Conexão com Wi-Fi;
- **void loop:**
 - Update das células de carga;
 - Desenvolvimento do vetor gradiente;
 - Comunicação HTTP para o envio dos dados para a base de dados;

O código programado para as células de carga está detalhado no Anexo C1.

Quando é referido o desenvolvimento do vetor gradiente, é o processo feito para que seja obtido a variação do movimento. A célula de carga mede o peso do animal, ou seja, quanto mais perto ele estiver de uma das células maior será o valor dessa. Uma vez que o peso do animal não varia, quando é feita a subtração do valor atual com o anterior, o peso deste é basicamente anulado, ficando apenas a variação do peso que a célula está a medir. Essa variação das medidas indica a deslocação do animal e quando é feito o módulo da variação destas, é indicado se houve muito ou pouco movimento, uma vez que se este tiver parado não vai haver variação entre as células. Este cálculo das variações é a soma das variações das 3 células em conjunto. Assim é quantificado o movimento e não a sua localização.

4.2.2 - Sensores - Arduíno UNO - WeMos D1 mini ESP8266

No anexo B2 é possível ver as ligações do sensor DHT22, do MLX90614 e da câmara LS-Y201 com o arduíno UNO e deste com o WeMos D1 mini ESP8266. A comunicação entre o arduíno UNO e o WeMos D1 mini utiliza o protocolo SPI (serial peripheral interface), que permite a criação de uma rede, usada para curtas distâncias. Assim, o arduíno faz a leitura de todos os sensores, processa, envia esses dados para o WeMos D1 mini e este é que vai permitir que esses cheguem ao computador por wi-fi e sejam enviados para a base de dados. Esta última parte, tal como explicado na secção 4.2.1, é feita através da comunicação HTTP seguindo o protocolo TCP/IP.

O primeiro passo para a programação é a inclusão das bibliotecas necessárias para cada sensor e para os tipos de comunicação. É de salientar, que nesta parte da programação a câmara LS-Y201 não foi conectada para fazer a ligação por Wi-Fi. O primeiro motivo é que esta sozinha ocupa mais de 50% da memória do microcontrolador. O segundo é que esta tem no mínimo 12 kb em cada imagem e os ciclos de envio de informação estão programados para enviar apenas 32 em 32 bit, ou seja uma só imagem ia demorar mais de 300 ciclos a ser enviada, o que para além de tornar o sistema mais lento, não ia apresentar a qualidade de vigia desejada.

Na tabela 4.2 é possível ver as bibliotecas incluídas para a programação e comunicação dos sensores DHT22 e MLX90614 com o arduíno UNO e o WeMos D1 mini ESP8266.

Tabela 4.2 - Bibliotecas necessárias para a conexão do DHT22 e do MLX90614 com o arduíno UNO e o WeMos D1 mini ESP8266.

Biblioteca	Utilidade
ESP8266WiFi.h	Programação do módulo
ESP8266WiFiMulti.h	Possibilidade de registar diversas redes Wi-Fi
String.h	Manipular strings
ESP8266HTTPClient.h	Comunicação HTTP
SPISlave.h	Comunicação do arduino UNO com o WeMos D1 mini
DHT.h	Programação do sensor de temperatura e humidade
Adafruit_MLX90614.h	Programação do sensor de temperatura IR MLX90614

A estrutura do código foi a seguinte:

- Introdução das bibliotecas;
- Definição de funções;
- Declaração das variáveis;
- **void setup:**
 - Conexão com o Wi-Fi;
 - Leitura e envio dos valores medidos;
- **void loop:**
 - Comunicação HTTP para o envio dos dados para a base de dados;

O código programado está detalhado no anexo C2.

4.3 - Base de dados

Uma base de dados permite o armazenamento de um conjunto de dados de uma forma persistente. Estes estão relacionados entre si de maneira a criar um sentido na informação pretendida. Neste caso em específico o intuito da base de dados criada é armazenar todas as informações do animal, informações dos médicos do hospital e as informações que os sensores estão a recolher da jaula em que o animal se encontra. A organização da base de dados é feita em tabelas, para que possibilite os diferentes tipos de informação, mas ao mesmo tempo que seja possível interligar todas elas. Este tipo de armazenamento de dados e a sua organização vão permitir uma maior facilidade do envio da mesma para o site desenvolvido.

A base de dados do sistema chama-se animalcare e foi criada utilizando o phpMyAdmin (aplicação web livre), utilizando a linguagem SQL (código da base de dados no Anexo D). Para uma melhor perceção dos dados guardados foram criadas 9 tabelas diferentes. Primeiramente a do “utilizador”, neste caso os médicos, com alguns dados deste, como o seu nome, e-mail e contacto. No entanto, para que os donos possam visitar os seus animais de forma online terão de ter acesso à mesma aplicação que os médicos, com acesso apenas à câmara, mas é de igual forma necessário que estes tenham um registo. Assim, há uma tabela também para estes, a tabela “dono”, que guarda apenas o seu nome, morada e contacto. Todo o resto das tabelas são ligadas ao animal, uma vez que este é o foco principal. Existe então a tabela “paciente”, na qual é feita um novo registo sempre que algum animal entra na clínica, ela guarda informações como o nome, tipo de animal, raça, sexo, datas de consultas e vacinas. A tabela “histórico”

retrata, tal como o nome indica, o historial clínico do animal, direcionado para as cirurgias ou internamentos, ou seja, esta guarda, a data em que o paciente entrou no internamento, a data de saída, o tipo de tratamento que fez e tem ainda um espaço para observações que sejam importantes de registar. Por fim, as tabelas “jaula”, “comida”, “bebida”, “movimento” e “temperatura_humidade” registam a data, têm ligação ao animal, e guardam os valores que estão a ser recolhidos dos sensores colocados na jaula.

As informações entram e são atualizadas na base de dados através dos registos e perfis criados no site, será explicado mais à frente, e também a partir dos sensores colocados na jaula como referido no subcapítulo 4.2. Na imagem 4.2 está representado o diagrama UML da base de dados animalcare facilitando a perceção da estrutura e o comportamento da mesma.

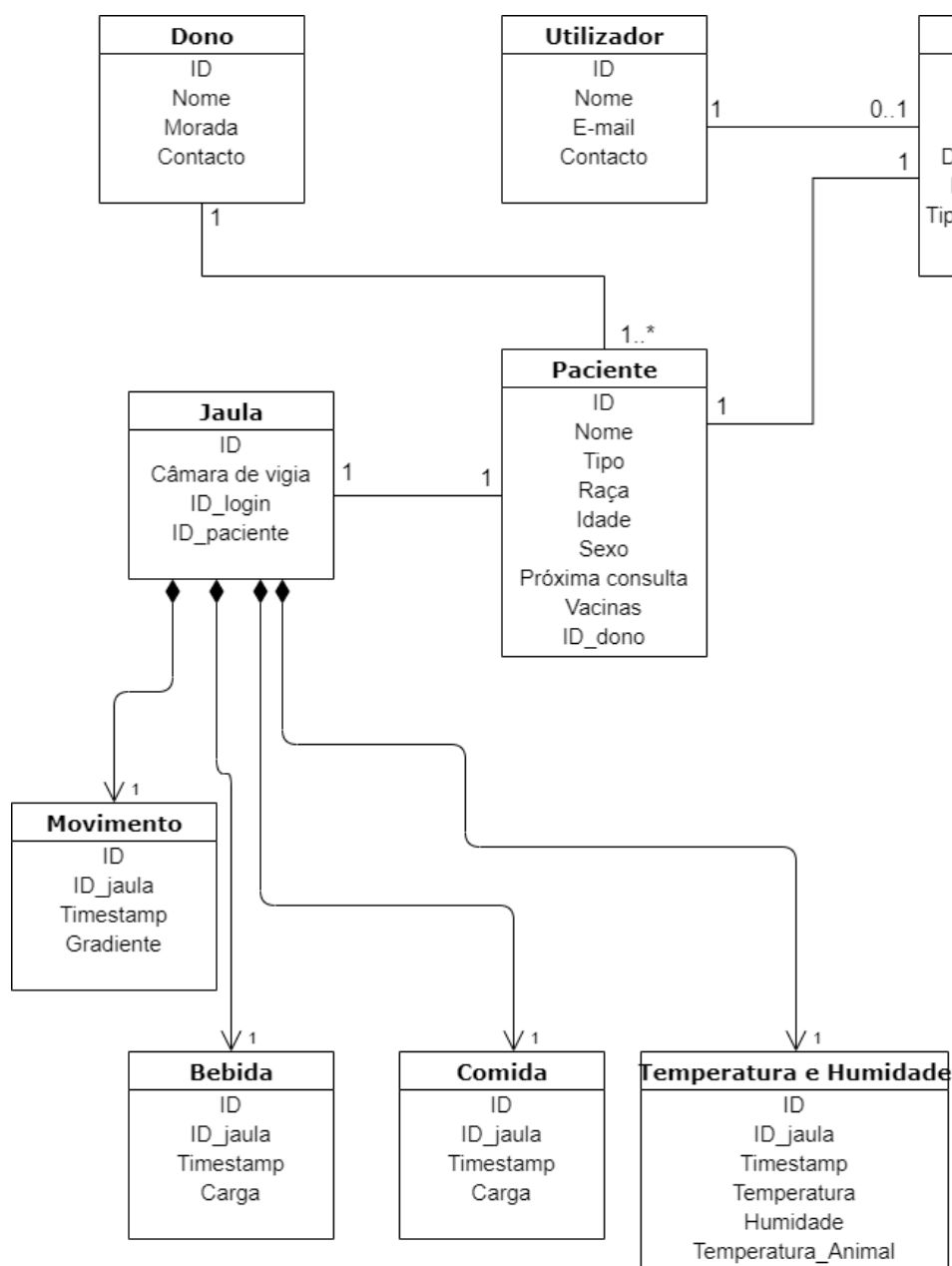


Figura 4.2 - Diagrama UML da base de dados animalcare.

4.3 - Comunicação base de dados - site

Como referido anteriormente, a base de dados quando criada, foi com o intuito de facilitar o armazenamento da informação e o envio da mesma para o site desenvolvido. No entanto, alguma da informação que chega à base de dados também vem do próprio site, ou seja, há uma troca de dados entres estes dois sistemas. A figura 4.3 apresenta um diagrama de fluxo dessas mesmas trocas. Resumidamente, há um utilizador (administrador), ao qual é permitido criar os registos dos pacientes e dos donos dos mesmos quando necessário. Esses registos são feitos no site e enviados, junto com todas as suas informações referentes, para a base de dados, onde são armazenados nas tabelas correspondentes. No sentido oposto, ou seja, da base de dados para o site as informações enviadas são referentes aos sensores que estão aplicados na jaula do animal.

A troca de informação entre os diferentes sistemas permite que qualquer alteração na informação do paciente seja logo atualizada também na base de dados, e no sentido inverso, sempre que entram valores novos na base de dados, são automaticamente atualizados no site, permitindo a qualquer momento consultar valores mais antigos.

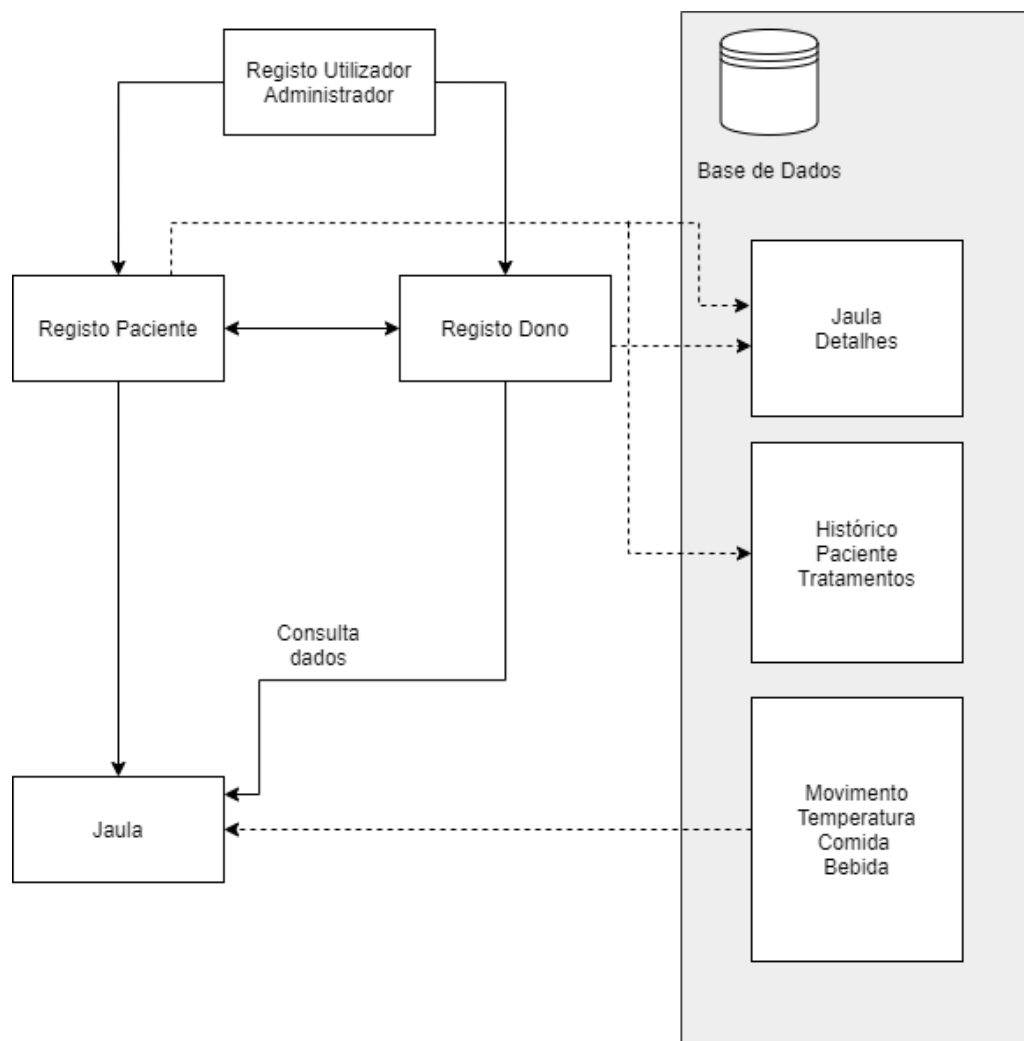


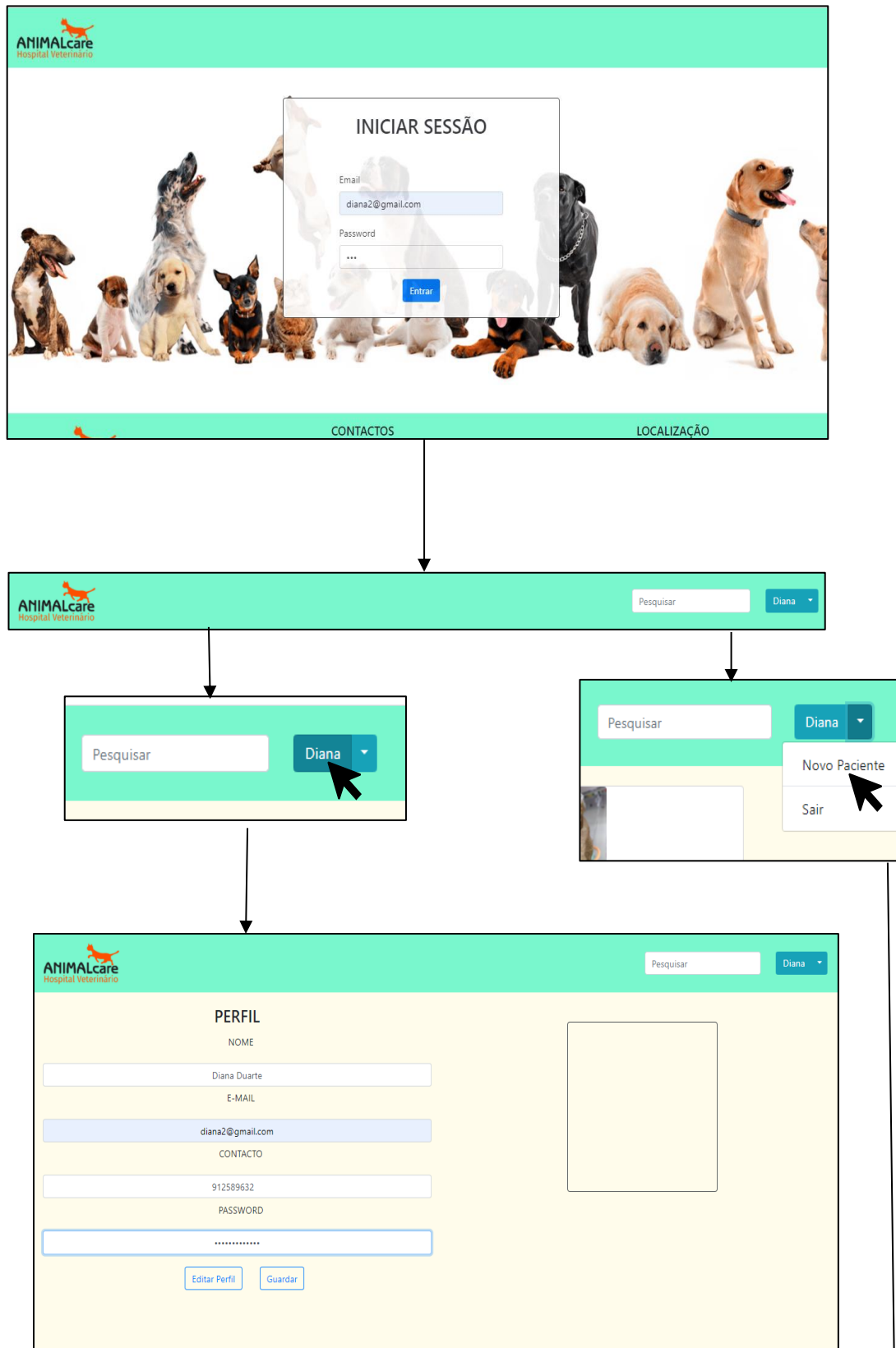
Figura 4.3 - Diagrama de fluxo.

4.4 - Site ANIMALcare

O final do sistema de monitorização passou pelo desenvolvimento de um site. Para a sua criação as linguagens utilizadas foram HTML, CSS e PHP, sendo a primeira para a estruturação da página, a segunda para criar uma estrutura dinâmica e a última para a sua funcionalidade, respetivamente. Ao fazer a sua criação desta forma, o HTML e o CSS são as linguagens visíveis e que o “cliente” utiliza, a parte do servidor é que tem acesso a todo o PHP, que é o que permite as funcionalidades do site, ou seja, o que “responde” às interações feitas pelo utilizador e permite a comunicação, neste caso, com a base de dados.

A principal função do site é, de uma forma simplista, receber os dados dos sensores que estão a monitorizar o animal mostrá-los, para que o médico em qualquer lugar do hospital possa aceder ao site e ver como esses mesmos parâmetros se encontram. No entanto, o site apresenta mais funções. Em todas as páginas existe um cabeçalho onde está indicado o nome do utilizador, se ele clicar no seu nome, dá-lhe acesso a uma página onde pode editar as suas informações do perfil. Tem, ainda no cabeçalho, ao lado do seu nome, uma seta que abre um dropdown que lhe dá acesso a fazer um registo de um novo paciente ou então a terminar a sua sessão (figura 4.4). A partir do momento em que o médico faz login no site, este entra numa página onde tem fotografias e dados básicos dos animais que está a seguir em internamento. Ao carregar na foto de um dos animais é aberta uma página onde é possível ver os gráficos obtidos com os valores que estão a ser medidos na jaula percebendo o seu comportamento e o seu estado de saúde remotamente (figura 4.5).

O site em si é simples e funcional, permite, como já referido anteriormente, que dados sejam enviados deste para a base de dados e vice-versa, controlando o animal à distância.



The image shows a web interface for 'ANIMALcare Hospital Veterinário'. The header is a teal bar containing the logo on the left, a search bar with the text 'Pesquisar' in the center, and a user profile dropdown with the name 'Diana' on the right. Below the header is a yellow section titled 'Registro do Paciente'. This section contains a vertical list of input fields for patient information: 'NOME', 'TIPO', 'RAÇA', 'IDADE', 'SEXO', and 'PRÓXIMA CONSULTA'. Each label is centered above its corresponding white input box. A black arrow on the right side of the image points from the top right towards the 'Registro do Paciente' section.

Figura 4.4 - Funções disponíveis no cabeçalho do site.

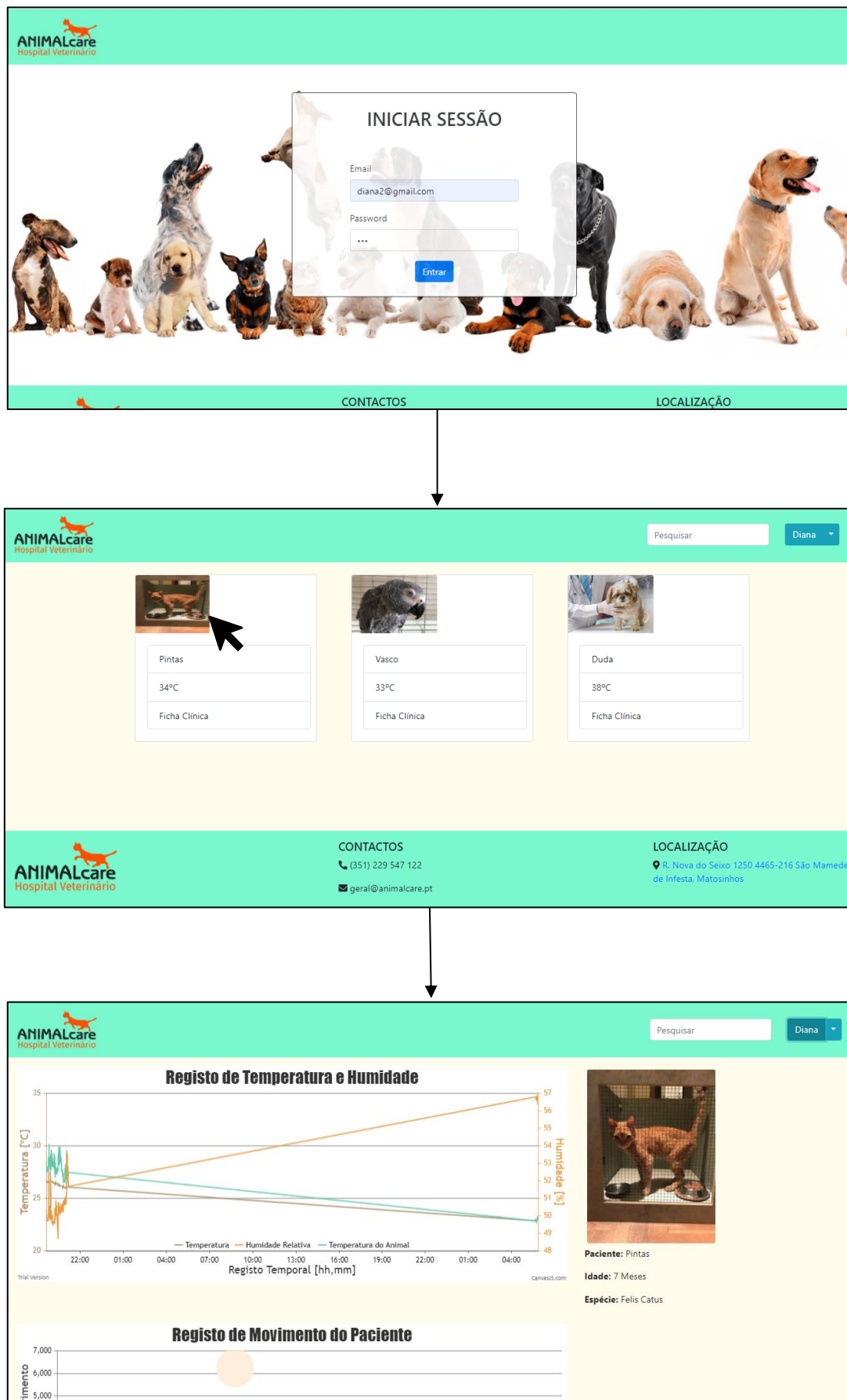


Figura 4.5 - Funcionalidades do site, disponíveis para o médico.

A figura 4.6 mostra, em esquema, a interação disponível no site, quando o utilizador é apenas um tutor de um animal internado e não um médico. Este apenas tem acesso a uma página onde estaria a câmara ligada para dar início à visita online.

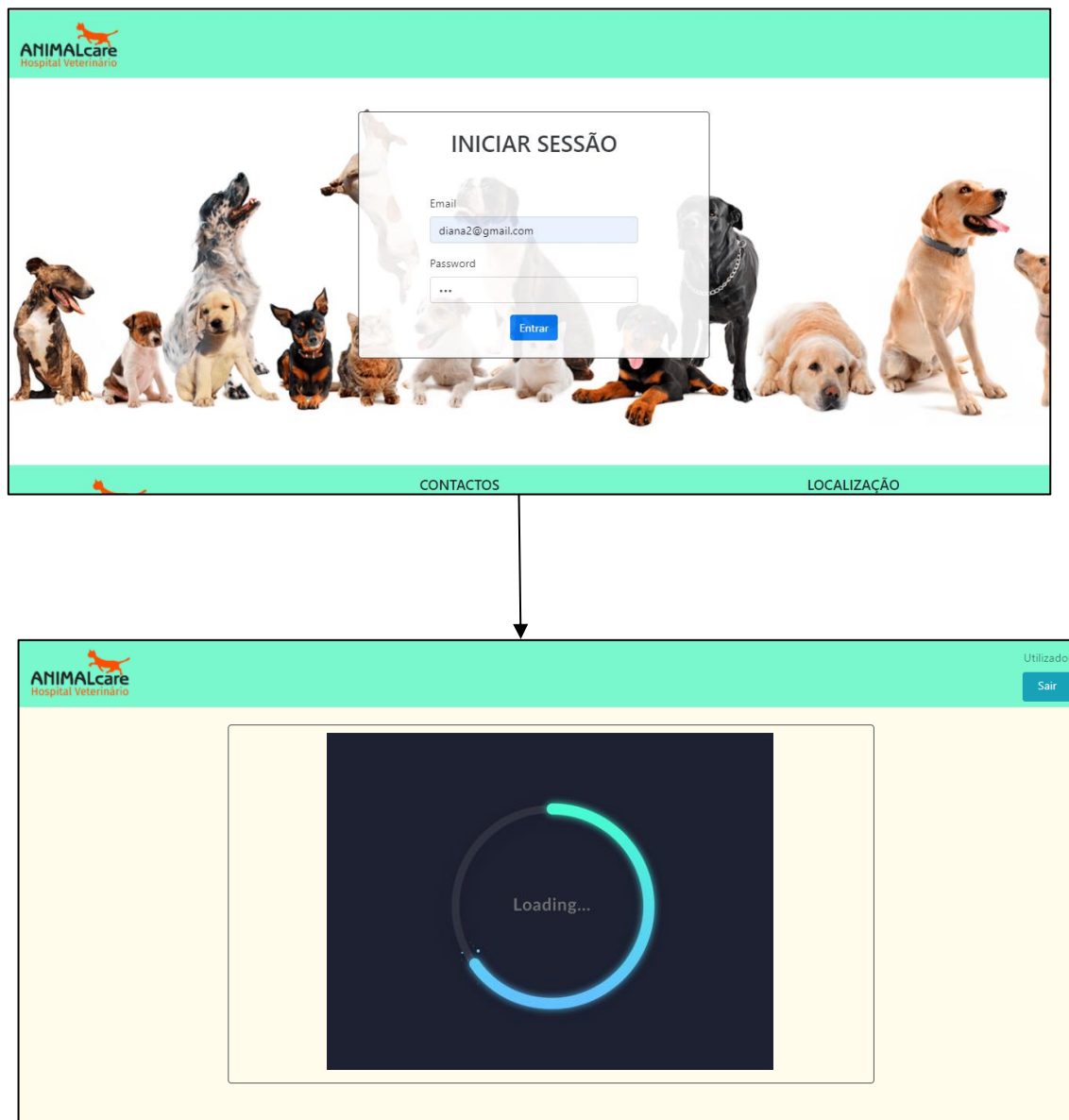


Figura 4.6 - Funcionalidades disponíveis no site quando se entra como tutor.

Capítulo 5

Resultados

Ao longo de toda a dissertação foi mencionado que o principal objetivo passava pela criação de um sistema de monitorização animal sem fios, que permitisse dar um maior conforto ao animal e uma maior tranquilidade ao corpo clínico que faz o seu controlo durante o internamento.

A primeira fase relacionou-se com o conforto animal e foi solucionada colocando todos os sensores para as medições na própria jaula, como é possível ver pela figura 4.1. Relativamente aos fios do sistema, que muitas vezes complicam a atividade do corpo clínico para obter medições, as comunicações dos sensores com o computador e a base de dados foram feitas por Wi-Fi, o que minimiza a sua quantidade no hardware. O facto da comunicação ser feita desta forma, permite à equipa clínica, durante a fase de monitorização, seguir todos os parâmetros medidos, observando apenas o monitor do computador onde o sistema está ligado e posteriormente, no site, observar a análise gráfica de todos os momentos da mesma.

Para efeito da validação do protótipo desenvolvido, foi necessário realizar a sua experimentação em ambiente real. Como tal, foi escolhido um animal de companhia, neste caso específico, um gato com aproximadamente 2,800 kg, e efetuou-se a sua monitorização ao longo de uma hora e meia. O foco inicial passou por proporcionar ao animal as melhores condições possíveis e o mais semelhante com o seu habitat natural durante todo o período de observação. Foi-lhe colocado uma porção de comida e bebida, cerca de 250g e 200ml, respetivamente.

O início da monitorização deu-se quando o animal já se encontrava no interior da jaula. Durante este período foi possível observar, no monitor do computador, todos os parâmetros definidos a serem medidos (figura 5.1), nomeadamente temperatura ambiente, humidade, temperatura animal, o seu movimento e por fim a quantidade de comida e bebida ingerida. É de salientar que, o sensor de temperatura IR MLX60914, está continuamente a enviar valores de temperatura, no entanto, apenas quando o animal se encontra no seu campo de visão é que este enviará o valor da sua temperatura corporal. Neste caso, como o sensor está colocado na perspetiva do comedouro e bebedouro, somente quando ele se aproxima dessa zona é que é medida a sua temperatura.

```

COM5
1 resultNew record created successfully
[HTTP] begin...
[HTTP] POST...
200
<!DOCTYPE html>
<html><body>
  <form action="/api/206_gsep.php" method="post">
    api_key: <input type="text" name="api_key"><br>
    <input type="submit">
  </form>
  1 resultNew record created successfully
  api_key: <input type="text" name="api_key">
  Question: IP address: 192.168.1.108
  [HTTP] begin...
  [HTTP] POST...
  Answer Sent
  200
  <!DOCTYPE html>
  <html><body>
    <form action="/api/206_gsep.php" method="post">
      api_key: <input type="text" name="api_key"><br>
      <input type="submit">
    </form>
    1 resultNew record created successfully
    api_key: <input type="text" name="api_key">
    Question: IP address: 192.168.1.108
    [HTTP] begin...
    [HTTP] POST...
    Answer Sent
    200
    <!DOCTYPE html>
    <html><body>
      <form action="/api/206_gsep.php" method="post">
        api_key: <input type="text" name="api_key"><br>
        <input type="submit">
      </form>
      1 resultNew record created successfully
  
```

Figura 5.1 - Monitor de visualização dos valores dos sensores.

Os dados recolhidos são apresentados no monitor e enviados para a base de dados em intervalos de 2 em 2 segundos. Posteriormente, através de uma função desenvolvida no site, os valores recolhidos são representados graficamente numa das páginas, de forma a facilitar a sua análise.

Os gráficos resultantes da monitorização do animal são, então, apresentados numa página do site. Um dos gráficos engloba a temperatura animal, a temperatura ambiente e a humidade, no entanto é possível seleccionar e ver cada um destes individualmente. O movimento e a quantidade de comida e bebida ingerida estão representados em separado, fazendo assim o total de quatro gráficos apresentados em cada monitorização realizada. Neste caso em específico, os resultados estão apresentados, em separado, nas figuras 5.2, 5.3, 5.4 e 5.5.

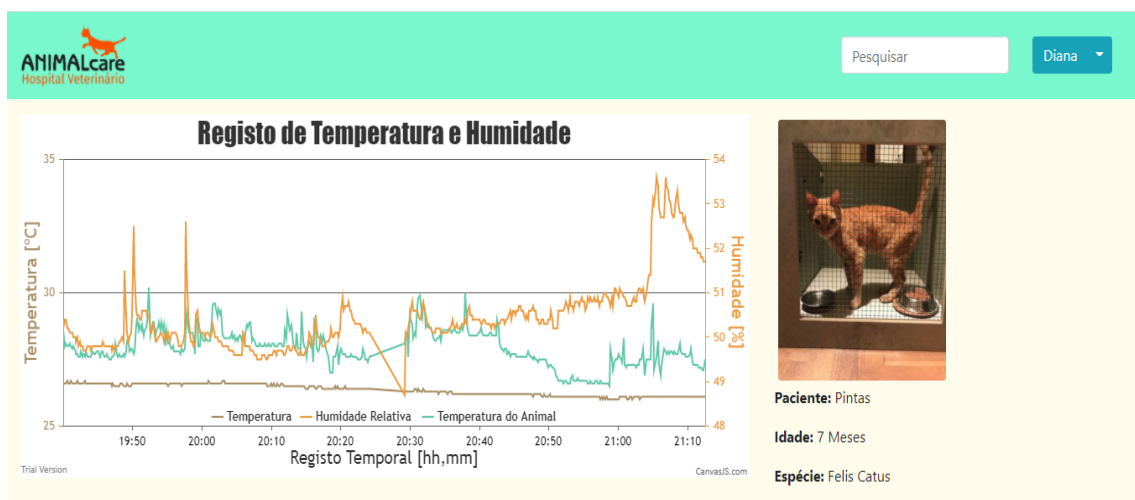


Figura 5.2 - Gráfico obtido da temperatura animal (linha verde), da temperatura ambiente (linha castanha) e da humidade relativa (linha laranja).

No caso da temperatura ambiente (representada pela linha castanha), como é possível observar, não houve grande variação, manteve-se à volta dos 26°C aproximadamente, o que era de esperar uma vez que a jaula se manteve, durante toda a monitorização, no mesmo local. O mesmo se pode afirmar da humidade relativa (representada pela linha laranja) apresenta

apenas uns picos pontuais, devido ao ruído, encontrando-se sempre entre os 50 e os 53%, o que não causa qualquer tipo de diferença para o bem-estar animal. Por último, no caso da temperatura do animal (representada pela linha verde) os resultados já foram diferentes. É possível constatar que quando há um aumento da temperatura significa que o animal se encontra na zona do comedouro e do bebedouro, uma vez que o sensor térmico IR se encontra a apontar para essa zona (será possível confirmar esta afirmação nos gráficos da comida e bebida). No entanto, o valor máximo medido por este sensor rondou os 30°C o que é considerado um valor relativamente baixo para a temperatura de um gato. Uma vez que, a literatura indica que este sensor de infravermelhos é bastante preciso ($\pm 0.5^\circ\text{C}$), possivelmente o erro poderá ter ocorrido na sua calibração, local de medida e distância.

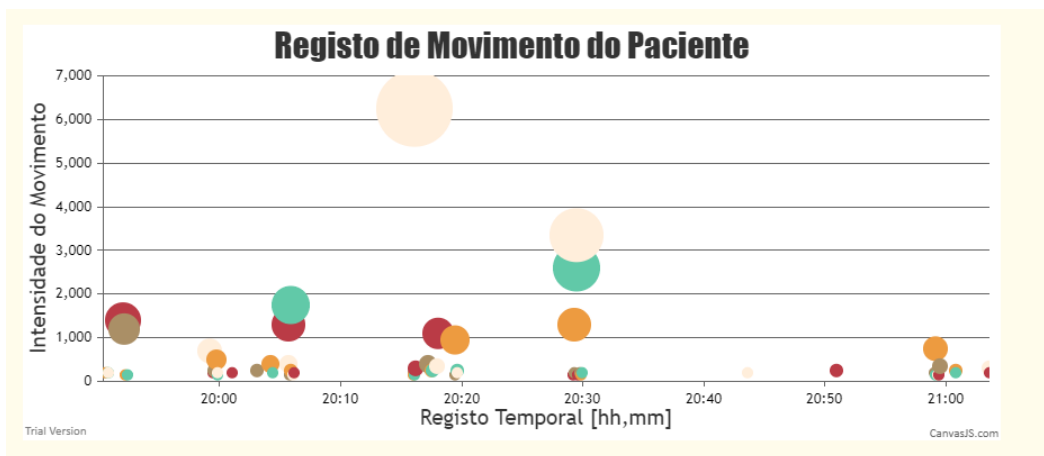


Figura 5.3 - Gráfico obtido pelo movimento do animal.

Como explicado nos capítulos anteriores, o objetivo das células de carga colocadas na base da jaula é monitorizar a existência de movimento. No gráfico, os círculos representados indicam o momento em que o animal se movimentou, e as cores representam as diferentes intensidades do mesmo, ou seja, se o movimento foi maior ou não. Por exemplo, no ponto das 20h20m, é perceptível que houve movimento, tal como nas 20h30m, no entanto, há valores de intensidade superiores no segundo ponto, respetivamente, o que significa que nesse mesmo ponto o deslocamento do animal foi maior, ele movimentou-se mais. Após uma análise, é possível concluir que o animal esteve relativamente sereno durante toda a monitorização, apresentando algum movimento, parte dele possivelmente justificado quando este se desloca para comer ou beber, o que pode ser verificado ao comparar com os gráficos a baixo.

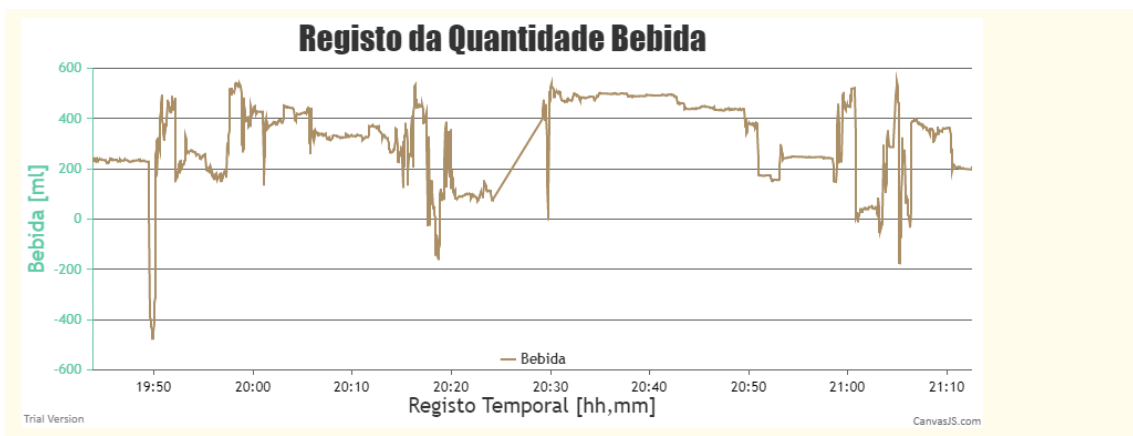


Figura 5.4 - Gráfico obtido pela célula de carga colocada no bebedouro do animal.

Relativamente ao gráfico da figura 5.4 do registo da quantidade de bebida ingerida pelo animal, é perceptível a existência de várias irregularidades ao longo do tempo. De facto, se correlacionarmos com os gráficos anteriores é possível justificar alguns dos picos ascendentes presentes, assumindo que o animal está a beber uma vez que houve movimento e a temperatura recolhida no sensor IR aumentou (pico das 19h50m por exemplo), no entanto, não é de interpretação fácil.

O esperado era que o animal ao ingerir água o valor medido fosse diminuindo, no entanto isso não se verificou, até pelo contrário. Um motivo válido para esta situação, e o qual não foi ponderado anteriormente, poderá ser o facto de este fazer peso sob o bebedouro quando está a beber, adulterando os valores de medição.

No panorama geral, através da interpretação do gráfico, compreende-se que embora existam intervalos em que os valores medidos são coerentes com o funcionamento da célula de carga, uma grande parte da medição encontra-se irregular. A justificação mais plausível para tal, passará por uma errada calibração da célula ou uma má conexão com os amplificadores.

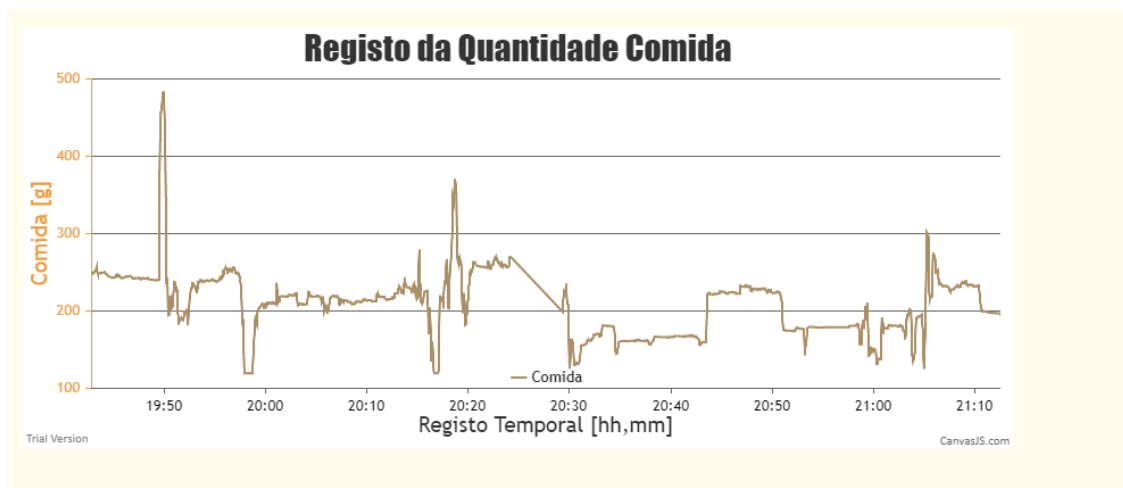


Figura 5.5 - Gráfico obtido pela célula de carga colocada no comedouro do animal.

No que diz respeito ao gráfico da figura 5.5 do registo da quantidade de comida ingerida pelo animal, observa-se uma coerência na diminuição do peso presente no comedouro. Inicialmente o comedouro continha 260g de comida e no final apresenta apenas 200g. Uma vez mais, é possível verificar a existência de pontos ascendentes que da mesma forma, podem ser justificados pela força do animal sobre o comedouro. De forma a suportar esta justificação, ao analisar os picos ascendentes percebe-se que imediatamente a seguir existe uma diminuição da quantidade de comida. Globalmente as medições apresentadas são coerentes com a diminuição da comida apresentada no comedouro, no entanto, os picos ascendentes e algumas das irregularidades estão, também, presentes no gráfico. Da mesma forma que a célula mencionada anteriormente, e uma vez que o método de calibração foi o mesmo, esta pode também apresentar erros por essa etapa não ter sido realizada corretamente.

Correlacionando o gráfico da temperatura, com o do movimento e o da quantidade de comida ingerida é possível confirmar algumas das conclusões retidas nas análises anteriores. A figura 5.6 mostra uma montagem dos três gráficos e assinalado estão os pontos em que é

coincidente afirmar que ele se moveu para ir comer e como é possível observar a temperatura medida pelo sensor de infravermelhos a aumentar.

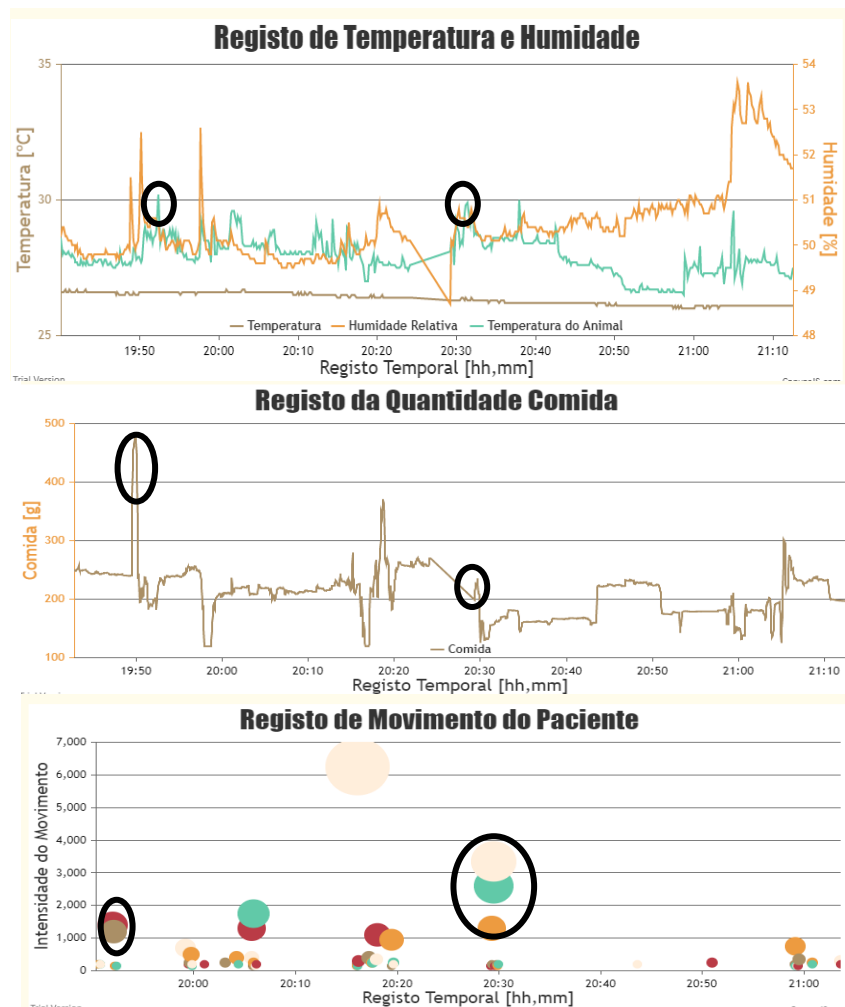


Figura 5.6 - Correlação entre os diferentes gráficos para mostrar momentos coincidentes.

De uma maneira geral, os resultados obtidos foram bastante positivos. Foi possível fazer toda a monitorização do animal à distância, enviar os dados para a base de dados e fazer posteriormente a sua análise no site.

Capítulo 6

Conclusão

A fase de recuperação pós-cirúrgica é dos momentos cruciais para o animal, uma vez que este se encontra em recuperação de potenciais alterações fisiológicas, induzidas pela anestesia. De forma a garantir uma boa estabilidade dessa fase, é necessário um controlo minucioso dos sinais vitais em pequenos períodos de tempo. Esta monitorização exige o uso de diversos equipamentos, sendo sempre necessário a intervenção de um médico ou enfermeiro veterinário e o manuseamento do animal. A utilização de equipamentos convencionais de monitorização com recurso a fios ou até mesmo ao simples facto de movimentar o animal para colocação de sensores podem causar muitas vezes um desconforto e mau-estar no paciente.

O estudo feito inicialmente, junto do corpo clínico veterinário, foi uma parte fundamental para perceber a necessidade hospitalar que existe nos dias de hoje e os problemas que a monitorização realizada atualmente traz.

A realização desta dissertação consistiu no desenvolvimento de um protótipo para responder a esta necessidade hospitalar. A partir das prioridades indicadas pelos profissionais de saúde foram estabelecidos os principais requisitos que a solução deveria apresentar, reduzir o número de fios, permitir uma monitorização sem causar incómodo no animal e com uma atenção especial para o custo. Assim, no protótipo a desenvolver pesou as prioridades, este incidiu em colocar os sensores na própria jaula e não no animal, para o maior conforto deste. A comunicação entre os sensores foi estabelecida por Wi-Fi para a redução dos fios e todo o material foi escolhido cuidadosamente com o intuito de assegurar um bom compromisso qualidade-preço.

O protótipo desenvolvido consistiu num conjunto de sensores colocados numa jaula, que recolhem os parâmetros movimento, temperatura animal, temperatura ambiente, humidade e quantidade de comida e bebida ingerida, e enviam simultaneamente para um computador e para uma base de dados via Wi-Fi que posteriormente envia para um site onde é feita a sua análise gráfica.

A validação foi positiva. O sistema de monitorização desenvolvido foi testado, não em ambiente hospitalar como era previsto, mas sim num ambiente familiar para o animal, no entanto os métodos de teste foram os mesmos e foi possível verificar o seu funcionamento. Os resultados foram coerentes e informativos sobre o estado do paciente embora haja ainda alguns problemas a resolver.

O sistema desenvolvido respondeu, em grande parte, aos objetivos propostos, permite a monitorização à distância, sem interferir com o animal e com a facilidade de ter todos os dados armazenados e analisados num site.

6.1 - Custos do projeto

O projeto desenvolvido apresentou custos, uma vez que foi necessária a compra dos equipamentos. Um dos principais objetivos deste, era o baixo custo de produção para que fosse possível a sua aquisição por parte dos centros hospitalares veterinários. Na tabela 6.1 estão detalhados os componentes que foram necessários, as quantidades e o seu respetivo valor comercial. É de salientar, que todo o material foi adquirido em lojas portuguesas, sendo que, em alguns casos, estes podem ser encontrados a um valor mais económico se comprados on-line.

Tabela 6.1 - Preço dos componentes adquiridos.

COMPONENTES	QUANTIDADE	PREÇO [€]
Arduíno UNO	1	23,75
Módulo Wireless ESP8266 NodeMCU	1	9,50
WeMos D1 mini NodeMCU ESP8266	1	11,07
Célula de carga - 5Kg	3	6,77
Célula de carga - 500g	2	17,71
Amplificador HX711	5	5,29
Sensor de temperatura e humidade DHT22	1	3,69
Adafruit MLX90614	1	45,85
Câmara LSY201	1	34,80
TOTAL	14	210,84

O custo do material utilizado foi de 210,84€, o que permite afirmar que o objetivo de conseguir obter um sistema de monitorização de baixo custo foi realizado, uma vez que, um simples monitor para um menor número de parâmetros custa no mercado entre 1500 e 5000€.

6.2 - Trabalhos futuros

A validação desta dissertação pode ser melhorada, uma vez que houve partes do objetivo que não foram concluídas. A componente de vigilância do animal (a câmara), que no sistema desenvolvido não está a funcionar por comunicação Wi-Fi, pode ser desenvolvido e permitir, dessa forma que os donos visitem o animal de forma online. Ao evoluir nesta componente, o

melhor é optar por uma câmara mais recente do que a LS-Y201. Hoje em dia já existem câmaras que apenas começam a filmagem quando o movimento é detetado, o que neste caso era vantajoso para o médico de controlo.

A medição da quantidade de comida e bebida ingeridos também pode ser desenvolvida, optando por comedouros e bebedouros fixos na parede lateral e não na base, visto que pode haver ruído na leitura do peso, como aconteceu neste sistema.

Em relação aos resultados em geral e de forma a melhorar a sua apresentação gráfica é possível recorrer à utilização de condensadores ou filtros de processamento, de forma a reduzir todos os ruídos captados e apresentar um intervalo de valores mais apelativos ao utilizador.

Por fim, é ainda possível desenvolver o site criado, para que este seja capaz de receber os dados em tempo real e alertar o corpo clínico em caso de anomalias com o paciente.

Referências

- [1] “A importância dos animais na vida das pessoas,” CPT e CENVA, 27 Março 2019. [Online]. Available: <https://www.cptcursospresenciais.com.br/blog/a-importancia-dos-animais-na-vida-das-pessoas/>. [Acedido em Janeiro 2020].
- [2] P. Educação, “O papel dos animais na sociedade,” [Online]. Available: <https://www.portaleducacao.com.br/conteudo/artigos/veterinaria/o-papel-dos-animais-na-sociedade/29326>. [Acedido em 2020].
- [3] P. Azevedo e R. Campilho, “Vida Ativa,” Dezembro 2018. [Online]. Available: <https://www.vidaativa.pt/a/animais-de-estimacao/>.
- [4] J. M. Bright, “Monitoring vital signs in clinical and research animals,” *Curr. Sep.*, vol. 16, pp. 43-47, 1997.
- [5] L. Priya, R. Hariprasad, and R. Raghul, “Real time monitoring of vital signs using wireless technique,” *Proceeding IEEE Int. Conf. Green Comput. Commun. Electr. Eng. ICGCCEE 2014*, 2014.
- [6] R. Brugarolas *et al.*, “Wearable Heart Rate Sensor Systems for Wireless Canine Health Monitoring,” *IEEE Sens. J.*, vol. 16, no. 10, pp. 3454-3464, 2016.
- [7] L. S. G. Vilaça and M. M. da Silva, “Animals Tracking P03,” pp. 1-13, 2016
- [8] C. T. F. Garcia, C. D. Viana, and L. Z. T. de Bragas, “Modalidade do trabalho : Relato de experiência Evento : XIV Jornada de Extensão,” *Salão do ConhecimentoLuz, ciência, vida*, pp. 2-5, 2010.
- [9] - P. d. Educação, “Monitorização em Anestesia de pequenos animais,” [Online]. Available: <https://www.portaleducacao.com.br/conteudo/artigos/veterinaria/monitorizacao-em-anestesia-de-pequenos-animais/73207>. [Acedido em Dezembro 2019]
- [10] CHRISTIANSEN, S. B.; FORKMAN, B. Assessment of animal welfare in a veterinary context - A call for ethologists. *Journal Applied Animal Behaviour*, v. 106, n.4, p. 203-220, 2007. Disponível em: . DOI: 10.1016/j.applanim.2007.01.004
- [11] R. Osman *et al.*, “No Title,” pp. 446-461, 2017, doi: 10.18677/EnciBio.

- [12] McMILLAN, F. D. Mental health and well-being in animals. Boston: Blackweell Publishing, 2005.
- [13] "ANIMALcare," [Online]. Available: <https://www.animalcare.pt/pt/servios>. [Acedido em 2020].
- [14] N. Alexandre and M. Veterinária, "Anestesia volátil e monitorização anestésica Universidade de Évora," 2008.
- [15] A. N. A. Carolina and D. A. Silva, "Faculdade de Medicina Veterinária," 2016.
- [16] L. Alex, "Anestesia Veterinária," [Online]. Available: <https://anestesiaveterinaria.weebly.com/importacircncia-damonitorizaccedilatildeo.html>. [Acedido em Dezembro 2019].
- [17] "Papers in conference."
- [18] Mazzaferro, E. M. (2015). Perioperative evaluation of the critically ill patient. In D. C. Silverstein & K. Hopper, Small animal critical care medicine (2nd ed., pp. 691-694). St. Louis, Missouri, USA: Saunders/Elsevier.
- [19] M. N. Andr, "MARTA NICEIA ANDRÉ PAULO AVALIAÇÃO DA DOR NA CIRURGIA E PÓS-OPERATÓRIO EM CADELAS SUBMETIDAS A MARTA NICEIA ANDRÉ PAULO AVALIAÇÃO DA DOR NA CIRURGIA E PÓS-," 2014.
- [20] Shih, A. (2015). Anesthesia and analgesia for geriatric and neonatal patients. In R. Hipple (Ed.), Proceedings of the NAVC Conference, Small Animal & Exotic edition, Book 1, January 17-21, Orlando, Florida. 29, pp. 90-92. Gainesville: NAVC.
- [21] Rozanski, E. A. & Rush, J. E. (2013). Monitoring critical care patients. In E. A. Rozanski & J. E. Rush, Small animal emergency and critical care medicine: A colour handbook (2nd ed., pp. 213-219). Boca Raton, Florida, USA: Mason Publishing.
- [22] Hackett, T. B. (2015). Physical examination and daily assessment of the critically ill patient. In D. C. Silverstein & K. Hopper, Small animal critical care medicine (2nd ed., pp. 6-10). St. Louis, Missouri, USA: Saunders/Elsevier.
- [23] Devey, J. J. (2012b). Early goal directed therapy. Proceedings of the 11th European Veterinary Emergency and Critical Care Society Congress: Care of the neurological animal, Small Animal Stream, 15-17 June, Barcelona, Spain (pp. 19-24). EVEECS.
- [24] Breton, A. (2014c). Understanding the hospitalized senior pet. Proceedings of the 20th International Veterinary Emergency and Critical Care Symposium, 10-14 September, Indianapolis, Indiana (pp. 691-694). IVECCS.
- [25] Feller, B. K. (2013). Code blue: preventing perioperative complications. Proceedings of the 19th International Veterinary Emergency and Critical Care Symposium, 7-11 September, San Diego, California (pp. 693-696). IVECCS.
- [26] Rivera, A. M. (2011b). Physiological parameters: assessment, monitoring and intervention. Proceedings of the 10th European Emergency and Critical Care Congress: O2 4U2- the oxygen pathway, 10-12 June, Utrecht, The Netherlands (pp. 71-77). EVEECS.

- [27] Martin, D. D. (2015). Capnography in practice. In R. Hipple (Ed.), Proceedings of the NAVC Conference, Small Animal & Exotic edition, Book 1, 17-21 January, Orlando, Florida. 29, pp. 65-66. Gainesville: NAVC.
- [28] Humm, K. (2012). How to interpret arterial blood gas. Proceedings of the BSAVA Congress 2012: Nursing Scientific Proceedings, 11-15 April, Birmingham, UK (pp. 393-395). Quedgeley: BSAVA.
- [29] Read, M. (2012). Physiology and prevention of perianesthetic hypothermia. Proceedings of the 18th International Veterinary Emergency and Critical Care Symposium, 8-12 September, San Antonio, Texas (pp. 21-24). IVECCS.
- [30] Grubb, T. L., Jimenez, T. E. & Pettifer, G. R. (2015). Senior and geriatric patients. In K. A. Grimm, L. A. Lamont, W. J. Tranquilli, S. A. Greene & S. A. Robertson (Eds.), Veterinary anesthesia and analgesia: the fifth edition of Lumb and Jones (5th ed., pp. 988-991). Hoboken, New Jersey, USA : Wiley-Blackwell.
- [31] Salazar, V. (2012). Pain management in the emergency and critically ill patient. Proceedings of the 11th European Veterinary Emergency and Critical Care Society Congress: Care of the neurological animal ,Small Animal Stream, 15-17 June, Barcelona, Spain (pp. 13-18). EVEEC.
- [32] “Ordem dos médicos veterinários,” [Online]. Available: <https://www.omv.pt/>. [Acedido em Janeiro 2020].
- [33] M. Veterin, “UNIVERSIDADE FEDERAL DO PARANÁ SETOR PALOTINA ATIVIDADES DO ESTÁGIO SUPERVISIONADO OBRIGATÓRIO Área : Clínica Médica e Cirúrgica de Pequenos Animais Aluno : Luana Vanessa Schons Supervisor : Aline de Marco Viott Orientador : Fabiano Brito dos Santos,” 2013.
- [34] A. Patil, C. Pawar, N. Patil, and R. Tambe, “Smart health monitoring system for animals,” *Proc. 2015 Int. Conf. Green Comput. Internet Things, ICGCIoT 2015*, pp. 1560-1564, 2016, doi: 10.1109/ICGCIoT.2015.7380715.
- [35] Avalia, Q. D. E. (2010). Prestação de cuidados a animais na perspectiva de uma ciência melhor.
- [36] L. M. Pires, “Redes de Sensores sem fios : Aplicações para Supervisão e Monitorização de Infraestruturas,” no. October, 2017, doi: 10.13140/RG.2.2.26495.92324.
- [37] E. Iot et al., The Three Software Stacks Required for IoT Architectures. Eclipse, 2016.
- [38] Meus animais, “Identificação animal por microchip.” [Online]. Available: <https://meusanimais.com.br/identificacao-animal-microchip/>. [Accessed: Dez-2019].
- [39] Identipet, “No Title.” [Online]. Available: <http://www.identipet.com/product/bio-thermo-temperature-sensing-microchip/>. [Accessed: Dez-2019].
- [40] VELASQUES RASTREL, “No Title.” [Online]. Available: www.velasquesrastrel.com.br/animais.php. [Accessed: Dez-2019]
- [41] WTEC, 2018. [Online]. Available: https://www.wtec.pt/prod/117/monitor_pm10a_vet_ultraslim?fbclid=IwAR3to4dq_hpGC_zCbc5U3K2scaP9wAueCLWZLqNJePFaRcTt6Q2wwwxWRPf0. [Acedido em Janeiro 2020].

- [42] “LUTECH VETERINARY INDUSTRIES, INC.,” 2018. [Online]. Available: https://www.lutechvet.com/datalys-v7-veterinary-monitor.html?fbclid=IwAR3_B5cFa9_00JRNzhzOQVBnzZu8THlMKdG8TjzCNe7DklG50pYLzHyZCcA. [Acedido em 2019].
- [43] “ferox,” 2018. [Online]. Available: http://store.ferox.com.br/monitor-multiparametrico-fx4000-p73?fbclid=IwAR1ONixUc70glQ14S3f06TiGKaEnKX5_l3hd1UQwl53Er7s3r3buepB7GjU. [Acedido em 2019].
- [44] Conect, “Medical Expo,” [Online]. Available: <https://www.medicalexpo.com/pt/prod/contec-medical-systems/product-68095-599589.html>. [Acedido em Dezembro 2019].
- [45] “Zooplus,” 2020. [Online]. Available: https://www.zooplus.pt/shop/caes/caixas_transporte_cao/gps/737824. [Acedido em Janeiro 2020].
- [46] PetPace, “No Title.” [Online]. Available: <http://petpace.es/>. [Accessed: Dez-2019].
- [47] PetPace, “SMART SENSING COLLAR.” [Online]. Available: <https://petpace.com/smart-sensing-collar/>. [Accessed: Dez-2019].
- [48] “Veterinary Practice News,” 28 Junho 2015. [Online]. Available: https://www.veterinarypracticenews.com/Voyce-Pro-Offers-Remote-Monitoring-of-Dogs-Health/?fbclid=IwAR2DSDWZk49NKqDUhuQpnKQqxwWVpgLhYaLYT1X1VthStdQWiHTo_ymzpq8. [Acedido em Janeiro 2020].
- [49] Voyce, “No Title.” [Online]. Available: <http://www.voyce.com/>. [Accessed: Dez-2019].
- [50] C. V. Data, “New Companion Animal Remote Health Monitoring Solution Strong Economicazxs per Collar.” pp. 4-7, 2018
- [51] Kiestarter, “Buddy.The Dog Collar Reimagined.” [Online]. Available: <https://www.kickstarter.com/projects/squeakerdogs/buddy-the-dog-collar-reimagined>. [Accessed: Dez-2019].
- [52] “PetNet,” [Online]. Available: https://www.petnet.io/pages/smartfeeder?fbclid=IwAR29DBV_mX3_1Cmlg_Hm4aeqYDtbJjXTSIPIQgvBz600oFq4KBZlwm42iVc. [Acedido em 2020].
- [53] “zooplus,” [Online]. Available: https://www.zooplus.pt/shop/caes/comedouro_bebedouro_cao/comedouros_automaticos_fontes_cao/outros_produtos/672442. [Acedido em Janeiro 2020].
- [54] T. A. Shinde and J. R. Prasad, “IoT based Animal Health Monitoring with Naive Bayes Classification,” vol. 4, no. cPGCON, pp. 8104-8107, 2017.
- [55] R. Pi, “Development of IoT Based Smart Animal Health Monitoring System using,” pp. 24-31, 2018.
- [56] P. Khatate, A. Savkar, and C. Y. Patil, “Wearable Smart Health Monitoring System for Animals,” 2018 2nd Int. Conf. Trends Electron. Informatics, no. lcoei, pp. 162-164, 2018.
- [57] D. C. -. M. University, “MEDEA,” 2013. [Online]. Available: <http://medea.mah.se/2013/04/arduino-faq/>. [Acedido em 07 2020].

- [58] E. Systems, “ESP8266EX datasheet,” 2020.
- [59] Sparkfun, “Sparkfun start something,” [Online]. Available: <https://learn.sparkfun.com/tutorials/what-is-an-arduino>. [Acedido em junho 2020].
- [60] G. Oliveira, “MasterWalker,” 2017. [Online]. Available: <https://blogmasterwalkershop.com.br/embarcados/nodemcu/nodemcu-uma-plataforma-com-caracteristicas-singulares-para-o-seu-projeto-iot/> . [Acedido em 2020].
- [61] G. Bauermeister, “Wireless e IoT,” filipeflop, 22 março 2018. [Online]. [Acedido em julho 2020].
- [62] AOSONG, “Temperature and humidity module”.
- [63] Melexis, “MLX90614 family,” 2019.
- [64] LinkSprite, “LinkSprite JPEG Color Camera,” 2010. [Online]. Available: <https://www.sparkfun.com/datasheets/Sensors/Imaging/1274419957.pdf>. [Acedido em 2020].
- [65] Sparkfun, “Sparkfun,” 2016. [Online]. Available: <https://learn.sparkfun.com/tutorials/getting-started-with-load-cells/all> . [Acedido em 2020].
- [66] Saddam, “Circuit Digest,” março 2017. [Online]. Available: <https://circuitdigest.com/microcontroller-projects/arduino-weight-measurement-using-load-cell>. [Acedido em 2020].
- [67] M. G. Straub, “USINAINFO,” setembro 2019. [Online]. Available: <https://www.usinainfo.com.br/blog/balanca-arduino-com-celula-de-peso-e-hx711-tutorial-calibrando-e-verificando-peso/> . [Acedido em 2020].
- [68] “MDN web docs,” [Online]. Available: <https://developer.mozilla.org/pt-BR/docs/Web/HTTP/Overview>. [Acedido em 2020].
- [69] E. Alecrim, “InfoWESTER,” 2011. [Online]. Available: <https://www.infowester.com/memoria.php>. [Acedido em junho 2020].

ANEXO A

 Inquérito realizado nos centros i3s, ANIMALcare E HVUTAD

Estudo de Caso - Funcionamento de um Sistema Wi-Fi para a monitorização e alerta de sinais vitais em animais na fase pós-cirúrgica

Sou a Diana Duarte, aluna do Mestrado de Engenharia Biomédica e este inquérito tem como objetivo recolher informação para a realização da Dissertação de Mestrado a efetuar na Faculdade de Engenharia da Universidade do Porto, sobre supervisão e parceria do orientador Professor Joaquim Gabriel e co-orientador Professor Luís Antunes.

Este questionário baseia-se na melhoria do funcionamento do Sistema de Recobro dos Hospitais Veterinários. A solução por nós apresentada é o desenvolvimento de um sistema sem fios, disponível para telemóvel e/ou computador, onde seja possível monitorizar sinais vitais e outras características do paciente em tempo real, permitindo assim a deteção e alerta imediato de mudanças no estado do animal, na fase de recobro. A comunicação direta entre os sensores colocados no animal (coleira) ou colocados na própria habitação, com o software posteriormente implementado no hospital, alerta de imediato o responsável de saúde, pondo assim um fim a erros decisivos na sobrevivência do paciente e aumentando a eficácia dos Serviços de Recobro no Hospital.

Os dados fornecidos são exclusivamente utilizados para fins de investigação científica. Agradeço, desde já, o seu contributo!

Nome:
Área Clínica:

1. O controle do animal na fase pós-cirúrgica pode ser crucial para o mesmo, devido aos efeitos de fármacos como a anestesia. Acha importante a criação de um sistema onde, à distância, seja possível controlar os parâmetros mais importantes para a recuperação do animal?
2. Quais considera os maiores problemas na monitorização da fase de recobro?
3. Nesta fase, que parâmetros acha mais importante monitorizar?
4. Que equipamentos/sistemas já existem disponíveis para esta monitorização?

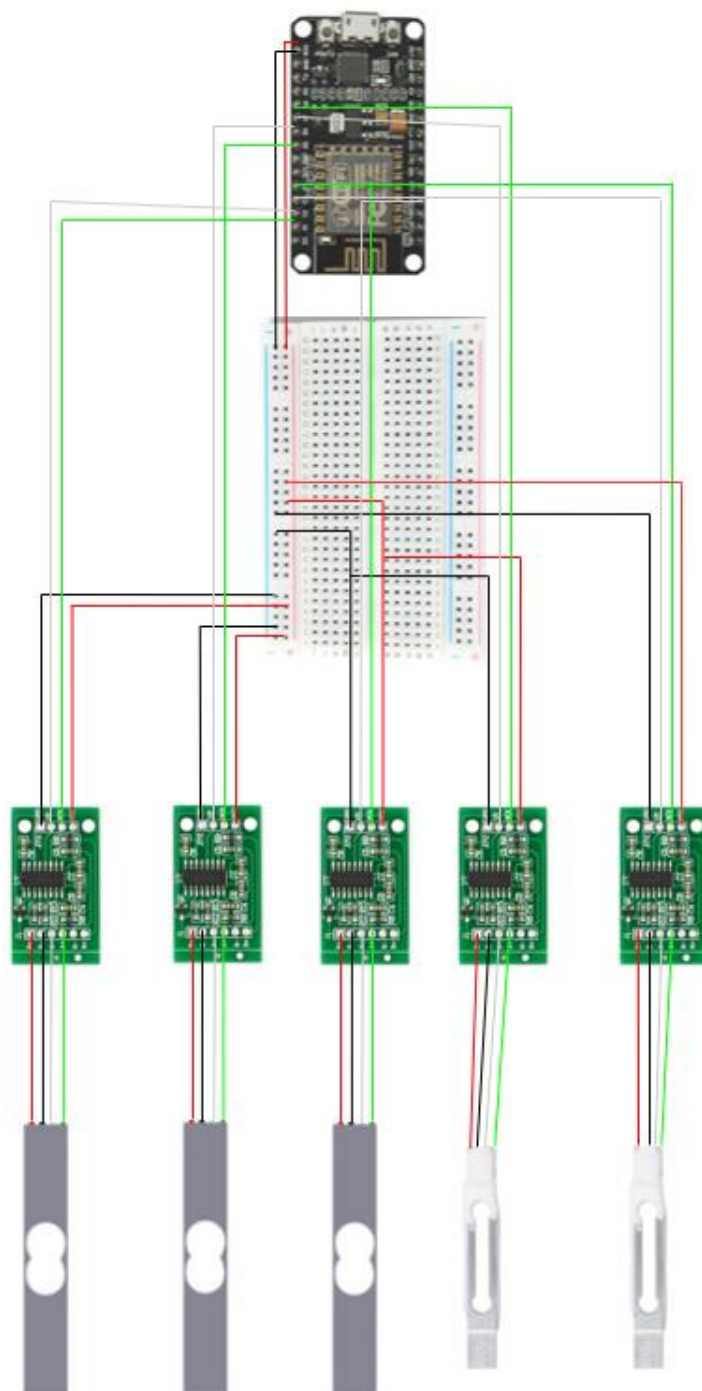
5. Nos equipamentos já existentes, que problemas e limitações existem? Como acha que poderiam ser resolvidos?
6. Das necessidades que essa monitorização apresenta, e com o que já há disponível, o que acha que deveria ser implementado? (controlo de medicação, quantidade de comida ingerida, ect)
7. Quais as vantagens e mais-valias que esta solução traria ao Serviço de Recobro?
8. Acha que esta solução seria útil comparando com os equipamentos já implementados?
9. Implementando este sistema, quais as espécies em que prevê maior utilização?
10. Qual a possibilidade da aplicação desta tecnologia noutras áreas da medicina veterinária?
11. De acordo com todas as necessidades, quanto estava disposto a pagar por um sistema destes?

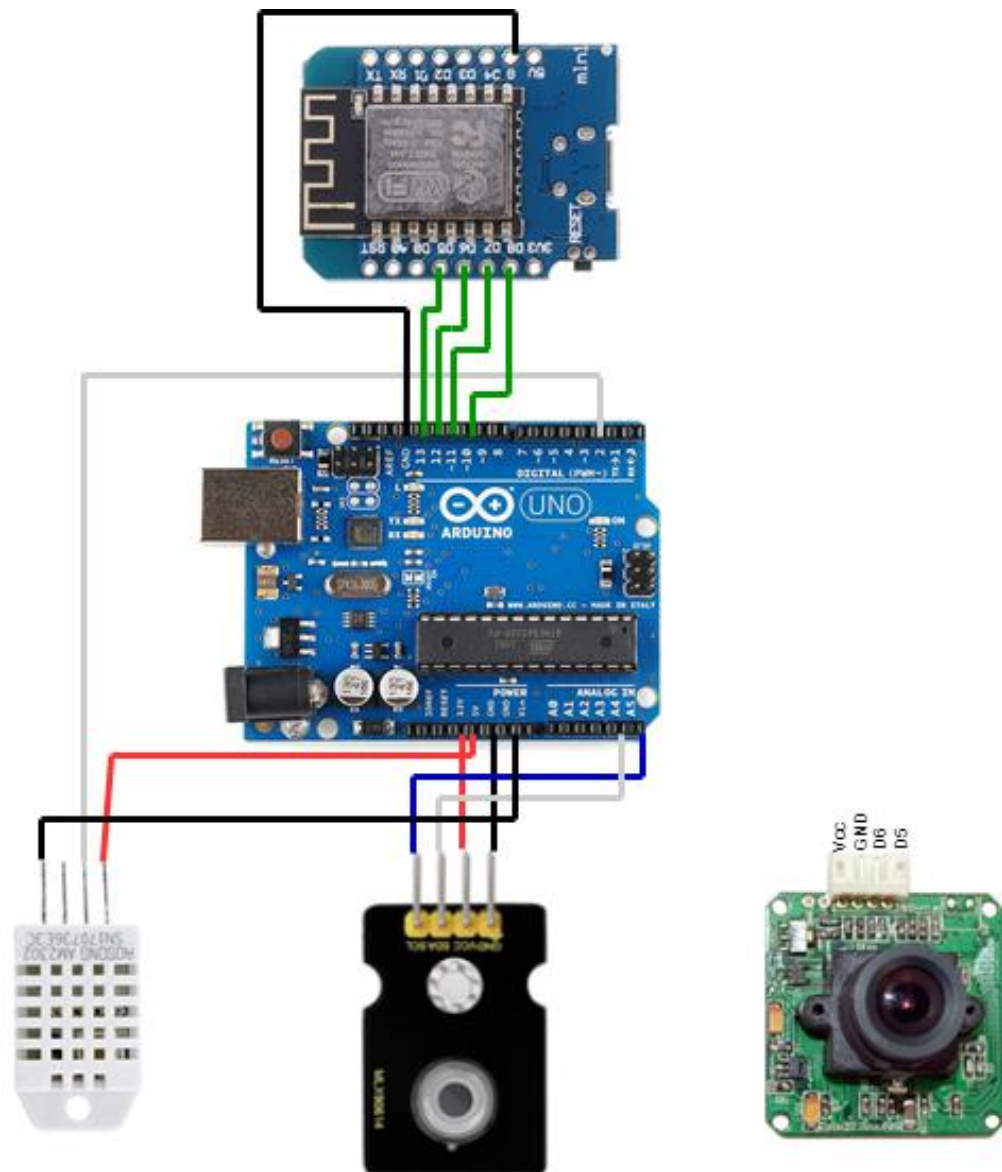
Caso queira deixar alguma sugestão ou informação que lhe tenha surgido ao longo deste questionário, no sentido de melhorar a ideia ou até mesmo o dispositivo, pode-o fazer aqui:

ANEXO B

Esquemáticos

B1 - Células de carga - ESP8266 NodeMcu



B2 - DHT22, MLX90614 - Arduino UNO - WeMos D1 mini

Código Arduino

C1 - Células de carga - ESP8266 NodeMcu

```

1  #include <ESP8266WiFi.h>
2  #include <ESP8266WiFiMulti.h>
3  #include <string.h>
4  #include <ESP8266HTTPClient.h>
5  #include <HX711_ADC.h> //versao 1.1.5
6  #include <stdlib.h>
7
8  //Configuracoes WIFI
9  const char* ssid    = "Vodafone-B425D6";
10 const char* password = "b7mez7uBe3";
11 //Celulas de carga
12 HX711_ADC LoadCell_1(4, 5); //HX711 1
13 HX711_ADC LoadCell_2(12, 14); //HX711 2
14 HX711_ADC LoadCell_3(10, 9); //HX711 3
15 HX711_ADC LoadCell_4(12, 14); //HX711 4
16 HX711_ADC LoadCell_5(4, 5); //HX711 5
17
18 long t;
19
20 const int led = 13;
21
22 float peso_ant[3], peso_atu[3];
23 float quant_mov=0;
24 uint8_t print_check=0;
25
26 String aux;
27 uint8_t volatile check=0;
28
29 void setup_celulacarga(){
30   float calValue_1;
31   float calValue_2;
32   float calValue_3;
33   float calValue_4;
34   float calValue_5;
35   calValue_1 = 696.0; //valor de calibração da célula 1
36   calValue_2 = 733.0; //valor de calibração da célula 2
37   calValue_3 = 733.0; //valor de calibração da célula 3

```

```

38 calValue_4 = 543.0; //valor de calibração da célula 4
39 calValue_5 = 709.0; //valor de calibração da célula 5
40
41 LoadCell_1.begin();
42 LoadCell_2.begin();
43 LoadCell_3.begin();
44 LoadCell_4.begin();
45 LoadCell_5.begin();
46 long stabilisingtime = 2000; //tempo para a tarar a balança
47 byte loadcell_1_rdy = 0;
48 byte loadcell_2_rdy = 0;
49 byte loadcell_3_rdy = 0;
50 byte loadcell_4_rdy = 0;
51 byte loadcell_5_rdy = 0;
52 while ((loadcell_1_rdy + loadcell_2_rdy + loadcell_3_rdy + loadcell_4_rdy
53 loadcell_5_rdy) < 5) {
54 //executa inicialização, estabilização e tara, ambos os módulos
55 if (!loadcell_1_rdy) loadcell_1_rdy = LoadCell_1.startMultiple(stabilisingtime);
56 if (!loadcell_2_rdy) loadcell_2_rdy = LoadCell_2.startMultiple(stabilisingtime);
57 if (!loadcell_3_rdy) loadcell_3_rdy = LoadCell_3.startMultiple(stabilisingtime);
58 if (!loadcell_4_rdy) loadcell_4_rdy = LoadCell_4.startMultiple(stabilisingtime);
59 if (!loadcell_5_rdy) loadcell_5_rdy = LoadCell_5.startMultiple(stabilisingtime);
60 }
61 if (LoadCell_1.getTareTimeoutFlag()) {
62   Serial.println("Tare timeout, check MCU>HX711 no.1 wiring and pin designations");
63 }
64 if (LoadCell_2.getTareTimeoutFlag()) {
65   Serial.println("Tare timeout, check MCU>HX711 no.2 wiring and pin designations");
66 }
67 if (LoadCell_3.getTareTimeoutFlag()) {
68   Serial.println("Tare timeout, check MCU>HX711 no.3 wiring and pin designations");
69 }
70 if (LoadCell_4.getTareTimeoutFlag()) {
71   Serial.println("Tare timeout, check MCU>HX711 no.3 wiring and pin designations");
72 }
73 if (LoadCell_5.getTareTimeoutFlag()) {
74   Serial.println("Tare timeout, check MCU>HX711 no.3 wiring and pin designations");
75 }
76 LoadCell_1.setCalFactor(calValue_1); //valor de calibração definido anteriormente
77 LoadCell_2.setCalFactor(calValue_2);
78 LoadCell_3.setCalFactor(calValue_3);
79 LoadCell_4.setCalFactor(calValue_4);
80 LoadCell_5.setCalFactor(calValue_4);
81
82 Serial.println("Startup + tare is complete");

```

```

83 }
84 void setup(void) {
85   pinMode(led, OUTPUT);
86   digitalWrite(led, 0);
87   Serial.begin(115200);
88   setup_celulacarga();
89
90   WiFi.begin("Vodafone-B425D6", "b7mez7uBe3");
91   while (WiFi.status() != WL_CONNECTED) { //Aguarda a conexão ao wi-fi
92     delay(500);
93     Serial.println("Waiting for connection");
94   }
95   Serial.println("HTTP server started");
96 }
97
98 void loop(void) {
99
100   LoadCell_1.update();
101   LoadCell_2.update();
102   LoadCell_3.update();
103   LoadCell_4.update();
104   LoadCell_5.update();
105   if (millis() > t + 2000) {
106     peso_ant[0]=peso_atu[0];
107     peso_ant[1]=peso_atu[1];
108     peso_ant[2]=peso_atu[2];
109     peso_atu[0] = LoadCell_1.getData();
110     peso_atu[1] = LoadCell_2.getData();
111     peso_atu[2] = LoadCell_3.getData();
112     float com=LoadCell_4.getData();
113     float beb=LoadCell_5.getData();
114     quant_mov=max(100,abs(med_pond(peso_atu[0],peso_ant[0])+med_pond(peso_atu[1],
115     peso_ant[1])+med_pond(peso_atu[2],peso_ant[2]))*50);
116
117     Serial.print("quanto:");
118     Serial.println(quant_mov);
119
120     print_check=5;
121
122     if (print_check==5){
123
124       t = millis();
125
126       aux = "&mov=" + String(quant_mov,2) + "&com=" + String(com,2) + "&beb=" +
127       String(beb,2);

```

```

128 Serial.printf("Question:");
129 Serial.println(aux);
130 aux=String("api_key=2cba61b&type=1&id=1") + aux;
131
132 if((WiFi.status() == WL_CONNECTED)) {
133
134     uint8_t y=aux.length();
135     char cmd_char[y+1];
136     aux.toCharArray(cmd_char, y+1);
137     HTTPClient http;
138     http.setTimeout(65000);
139
140     Serial.print("[HTTP] begin...\n");
141
142     http.begin("http://192.168.1.73/tese/esp8266_prep.php"); //comunicação
143     com o endereço que eu quero para enviar pa base de dados
144
145     http.addHeader("Content-Type", "application/x-www-form-urlencoded",
146     false, true);
147     Serial.print("[HTTP] POST...\n");
148     //inicia conexão
149
150     int httpCode = http.POST(cmd_char); //envia o pedido
151     String payload = http.getString(); //obtem a resposta
152
153     Serial.println(httpCode);
154     Serial.println(payload);
155
156     http.end();
157 }
158
159 }
160 }
161 //check if last tare operation is complete
162 if (LoadCell_1.getTareStatus() == true) {
163     Serial.println("Tare load cell 1 complete");
164 }
165 if (LoadCell_2.getTareStatus() == true) {
166     Serial.println("Tare load cell 2 complete");
167 }
168 if (LoadCell_3.getTareStatus() == true) {
169     Serial.println("Tare load cell 3 complete");
170 }
171 if (LoadCell_4.getTareStatus() == true) {
172     Serial.println("Tare load cell 4 complete");

```

```
173 }  
174 if (LoadCell_5.getTareStatus() == true) {  
175     Serial.println("Tare load cell 5 complete");  
176 }  
177 }  
178  
179 float med_pond(float a, float b){  
180     float med=(a+b)/2;  
181     if (med==0)  
182         return 0;  
183     return abs(a-b)/med;  
184 }  
185 float max(float a, float b){  
186     if (a<b)  
187         return b;  
188     else  
189         return a;  
190 }
```

C2 - DHT22, MLX90614 - Arduino UNO - WeMos D1 Mini

```

1  #include <ESP8266WiFi.h>
2  #include <ESP8266WiFiMulti.h>
3  #include <string.h>
4  #include <ESP8266HTTPClient.h>
5  #include <SPISlave.h>
6  #include <DHT.h>
7  #include <Adafruit_MLX90614.h>
8
9  char amostra[128];
10 char buff[128];
11 int lock=0;
12 void guardabuff(){
13     float * v = (float *) amostra;
14     String amb = String(v[0]);
15     String obj = String(v[1]);
16     String temp = String(v[2]);
17     String humi = String(v[3]);
18     sprintf(buff, "{\n\t\"MLX_Amb_C\":%s,\n\t\"MLX_Obj_C\":%s,\n\t\"
19     DHT22_Temp_C:%s,\n\t\"DHT22_Humi_RH\":%s\n}",
20
21     amb.c_str(),obj.c_str(),temp.c_str(),humi.c_str());
22 }
23 String aux;
24 uint8_t volatile check=0;
25
26 void setup() {
27
28     Serial.begin(9600);
29     // Serial.setDebugOutput(true);
30
31     Serial.println();
32     Serial.println();
33     Serial.println();
34
35     WiFi.begin("Vodafone-B425D6", "b7mez7uBe3");
36     while (WiFi.status() != WL_CONNECTED) { //Aguarda a conexão ao wi-fi
37         delay(500);
38         Serial.println("Waiting for connection");
39     }
40     memset(amostra,sizeof(amostra),0);
41
42     // data has been received from the master. Beware that len is always 32

```

```

43 // and the buffer is autofilled with zeroes if data is less than 32 bytes long
44
45 SPISlave.onData([](uint8_t * data, size_t len) {
46     String message = String((char *)data);
47     (void) len;
48     noInterrupts();
49     if (message.equals("Hello Slave!")) {
50         SPISlave.setData("Hello Master!");
51     } else if (message.equals("Are you alive?")) {
52         char answer[33];
53         sprintf(answer, "Alive for %lu seconds!", millis() / 1000);
54         SPISlave.setData(answer);
55     } else {
56         SPISlave.setData((char *)data);
57         strcpy(amostra, (char *) data);
58         //guardabuff();
59         char *parts;
60         parts=strtok(amostra, "_");
61         aux = "&obj=" + String(parts) + "&temp=" + String(strtok(NULL, "_")) +
62             "&humi=" + String(strtok(NULL, "_"));
63         Serial.println(aux);
64         aux=String("api_key=2cba61b&type=2&id=1") + aux;
65
66         Serial.printf("Question:");
67
68         check=1;
69     }
70     interrupts();
71     Serial.print("IP address: ");
72     Serial.println(WiFi.localIP());
73 });
74 // The master has read out outgoing data buffer
75 // that buffer can be set with SPISlave.setData
76 SPISlave.onDataSent([]() {
77     Serial.println("Answer Sent");
78 });
79 // status has been received from the master.
80 // The status register is a special register that bot the slave and
81 the master can write to and read from.
82 // Can be used to exchange small data or status information
83 SPISlave.onStatus([](uint32_t data) {
84     Serial.printf("Status: %u\n", data);
85     SPISlave.setStatus(millis()); //set next status
86 });
87 // The master has read the status register

```

```

88 SPISlave.onStatusSent([]() {
89     Serial.println("Status Sent");
90 });
91 // Setup SPI Slave registers and pins
92 SPISlave.begin();
93 // Set the status register (if the master reads it, it will read this value)
94 SPISlave.setStatus(millis());
95 // Sets the data registers. Limited to 32 bytes at a time.
96 // SPISlave.setData(uint8_t * data, size_t len); is also available with
97 the same limitation
98 SPISlave.setData("Ask me a question!");
99
100 Serial.println("");
101 Serial.print("IP address: ");
102 Serial.println(WiFi.localIP());
103 }
104 char str[]="api_key=2cba61b1";
105 void loop() {
106     // wait for WiFi connection
107     if((WiFi.status() == WL_CONNECTED) && check==1) {
108         check=0;
109         uint8_t t=aux.length();
110         char cmd_char[t+1];
111         aux.toCharArray(cmd_char, t+1);
112         HTTPClient http;
113         http.setTimeout(65000);
114
115         Serial.print("[HTTP] begin...\n");
116
117         http.begin("http://192.168.1.73/tese/esp8266_prep.php");
118         http.addHeader("Content-Type", "application/x-www-form-urlencoded",
119             false, true);
120
121         Serial.print("[HTTP] POST...\n");
122         // start connection and send HTTP header
123
124
125         int httpCode = http.POST(cmd_char); //Send the request
126         String payload = http.getString(); //Get the response payload
127
128         Serial.println(httpCode); //Print HTTP return code
129         Serial.println(payload); //Print request response payload
130
131         http.end();

```

132	}	
133		
134	}	

ANEXO D

Código Base de Dados

```
1 CREATE TABLE utilizador (  
2     id SERIAL PRIMARY KEY,  
3     nome varchar(255) NOT NULL,  
4     email varchar(255) UNIQUE NOT NULL,  
5     password varchar(255) NOT NULL,  
6     contacto integer NOT NULL,  
7     cargo integer DEFAULT 2,  
8     foto varchar(255) DEFAULT NULL  
9 );  
10  
11  
12 CREATE TABLE jaula (  
13     id SERIAL PRIMARY KEY,  
14     camara varchar(255) NOT NULL,  
15     id_login integer NOT NULL REFERENCES login(id),  
16     id_paciente integer NOT NULL REFERENCES paciente(id)  
17 );  
18  
19  
20 CREATE TABLE paciente (  
21     id SERIAL PRIMARY KEY,  
22     nome varchar(255) NOT NULL,  
23     tipo text NOT NULL,  
24     raça text NOT NULL,  
25     idade NUMERIC NOT NULL,  
26     sexo text NOT NULL,  
27 )
```

```

2          proxima_consulta text,
5
2          vacinas text NOT NULL,
6
2          id_jaula integer NOT NULL REFERENCES jaula(id),
7
2          id_dono integer NOT NULL REFERENCES dono(id)
8
2      );
9
3
0
3
1  CREATE TABLE dono (
3
2          id SERIAL PRIMARY KEY,
3
3          nome varchar(255) NOT NULL,
3
4          morada text NOT NULL,
3
5          contacto integer NOT NULL
3
6      );
7
3
8  CREATE TABLE historico (
3
9          id SERIAL PRIMARY KEY,
4
0          data_entrada DATE NOT NULL,
4
1          data_saida DATE NOT NULL,
4
2          tipo_de_tratamento text NOT NULL,
4
3          observacoes text,
4
4          id_paciente integer NOT NULL REFERENCES paciente(id),
4
5          id_utilizador integer NOT NULL REFERENCES utilizador(id)
4
6      );
7
4
8  CREATE TABLE movimento (

```

```
4      id SERIAL PRIMARY KEY,
9
5      id_jaula integer NOT NULL REFERENCES jaula(id),
0
5      data TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE
1      CURRENT_TIMESTAMP,
5
2      gradiente decimal
3
5  );
4
5
5
5
6
5  CREATE TABLE comida (
7
5      id SERIAL PRIMARY KEY,
8
5      id_jaula integer NOT NULL REFERENCES jaula(id),
9
6      data TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE
0
6      CURRENT_TIMESTAMP,
1
6      carga decimal
2
6  );
3
6
4
6  CREATE TABLE bebida (
5
6      id SERIAL PRIMARY KEY,
6
6      id_jaula integer NOT NULL REFERENCES jaula(id),
7
6      data TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE
8
6      CURRENT_TIMESTAMP,
9
7      carga decimal
0
7  );
1
7
2
```

```
7 CREATE TABLE temperatura_humidade (  
3  
7         id SERIAL PRIMARY KEY,  
4  
7         id_jaula integer NOT NULL REFERENCES jaula(id),  
5  
7         data TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE  
6  
7         CURRENT_TIMESTAMP,  
7  
7         temperatura decimal,  
8  
7         humidade decimal,  
9  
8  
0         temperatura_animal decimal  
8  
1 );
```