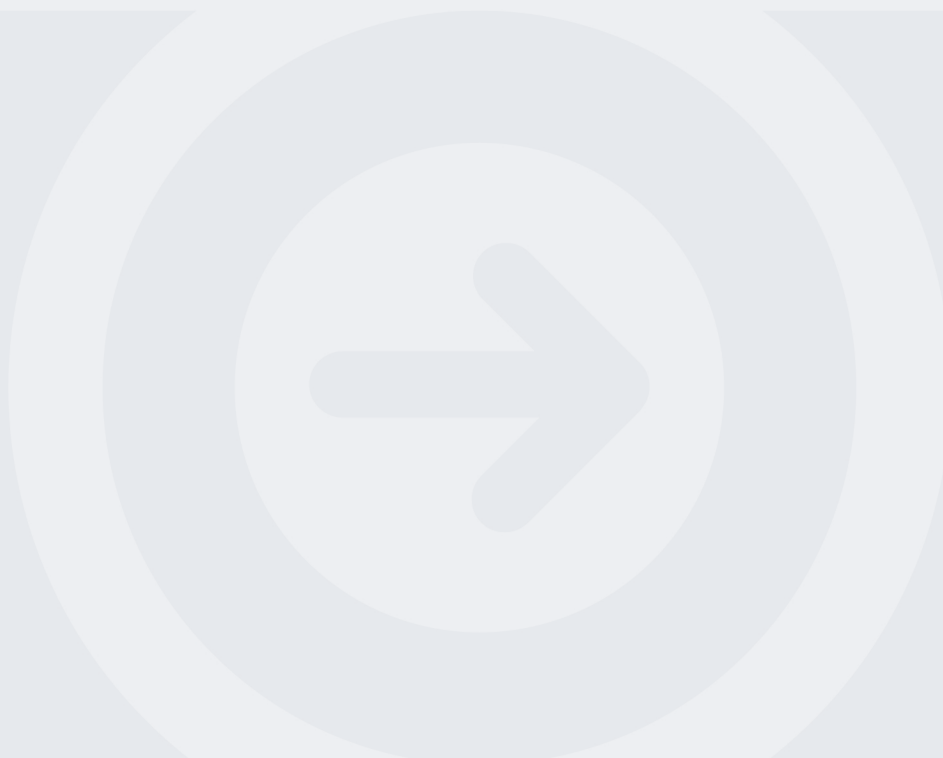


Monitorização do estado de saúde geral em comunidades isoladas

ANDRÉ FILIPE DOS SANTOS MOTA FERREIRA

dezembro de 2024



Monitorização do estado de saúde geral em comunidades isoladas

André Filipe dos Santos Mota Ferreira

**Dissertação para obtenção do Grau de Mestre em
Engenharia Biomédica**

Orientador: Prof. Dr. Joaquim Alves

Júri:

Presidente: Prof. Dra. Natércia Lima, Professora-Adjunta, Instituto Superior de Engenharia do Porto

Vogais:

Prof. Dr. Luís Coelho, Professor-Adjunto, Instituto Superior de Engenharia do Porto

Prof. Dr. Joaquim Alves, Professor-Adjunto, Instituto Superior de Engenharia do Porto

Resumo

A ideia para este trabalho nasceu devido ao quadro pandémico que ocorreu entre 2020 e 2022, onde fomos obrigados a ficar em casa e a manter o distanciamento social. Contudo, em locais onde o isolamento impera, e onde existem idosos que vivem sozinhos sem qualquer tipo de vigia regular, onde muitas vezes, a distância e as localizações remotas com os difíceis acessos, dificultam a chegada da ajuda, o que por vezes pode ser fatal. Tendo isto em conta, pretende-se, portanto, criar um sistema que permita a monitorização regular do estado de saúde geral destes idosos, contribuindo assim, para uma melhoria na sua segurança e qualidade de vida. Este projeto vem então oferecer uma solução para, nestes locais remotos e isolados, se conseguir manter uma monitorização diária do estado de saúde geral destes idosos. Cada idoso irá ter em sua casa um módulo de aquisição de dados, o que fará com que possam todos os dias registar os seus parâmetros fisiológicos de saturação de oxigénio, frequência cardíaca e temperatura corporal. No final da leitura, os valores serão enviados por Wi-Fi ou GPRS para um servidor na *Cloud*. A junta de freguesia da aldeia e/ou o quartel dos bombeiros mais próximo, poderão ter acesso a este servidor, e visualizar os dados em tempo real através da *interface* de utilizador. Caso algum idoso obtenha uma leitura crítica, é gerada uma notificação pop-up no ecrã a indicar qual o utente que obteve aquela leitura e a respetiva data e hora a que a mesma foi feita. Desta forma, seria possível enviar uma unidade médica de urgência à morada do idoso, de forma pró-ativa, evitando assim complicações do estado de saúde do mesmo. Também nestes casos, um ou mais familiares do idoso podem ser notificados via e-mail pelo sistema. Os alertas gerados devido a leituras críticas de cada idoso serão guardados numa base de dados dedicada, por forma a conseguir manter um histórico de cada idoso, permitindo assim o rastreio de uma eventual patologia. No final, foi então possível criar um sistema funcional que serve de prova de conceito para uma implementação futura de forma escalável.

Palavras-chave: IoT, Isolamento, Saúde, Monitorização

Abstract

The idea for this work was born due to the pandemic that occurred between 2020 and 2022, where we were forced to stay at home and maintain social distancing. However, in places where isolation prevails, and where there are elderly people who live alone without any kind of regular supervision, where often, the distance and remote locations with difficult access make it difficult for help to arrive, which can sometimes be fatal. Taking this into account, the aim is, therefore, to create a system that allows regular monitoring of the general health status of these elderly people, thus contributing to an improvement in their safety and quality of life. This project therefore offers a solution for daily monitoring of the general health status of these elderly people in these remote and isolated locations. Each elderly person will have a data acquisition module in their home, which will allow them to record their physiological parameters of oxygen saturation, heart rate and body temperature every day. Once the reading is complete, the values will be sent via Wi-Fi or GPRS to a server in the Cloud. The village parish council and/or the nearest fire station will be able to access this server and view the data in real time through the user interface. If an elderly person obtains a critical reading, a pop-up notification will be generated on the screen indicating which user obtained the reading and the respective date and time it was taken. In this way, it would be possible to send an emergency medical unit to the elderly person's address, proactively, thus avoiding complications to their health status. In these cases, one or more of the elderly person's family members can also be notified via email through the system. The alerts generated due to critical readings of each elderly person will be saved in a dedicated database, in order to maintain a history of each elderly person, thus allowing the tracking of any possible pathology. In the end, it was then possible to create a functional system that serves as a proof of concept for a future implementation in a scalable way.

Keywords: IoT, Isolation, Health, Monitoring

Índice

1	Introdução	1
1.1	Soluções equivalentes já desenvolvidas	2
1.1.1	Solução desenvolvida por alunos do Instituto Politécnico de Leiria	2
1.1.2	Solução desenvolvida por alunos da Faculdade de Medicina da Universidade do Porto e da Faculdade de Ciências da Universidade do Porto.	2
1.1.3	Solução desenvolvida por alunos e docente da Universidade SRM na Índia	3
2	Bases Científicas e Tecnológicas	5
2.1	Monitorização de parâmetros fisiológicos	5
2.1.1	Frequência Cardíaca	5
2.1.2	Saturação de Oxigénio	5
2.1.3	Foto pletismografia	6
2.1.4	Oximetria de pulso	7
2.2	IoT - Internet das coisas	7
2.2.1	IoT na Medicina	7
2.3	Sensor PPG baseado em ESP32	8
2.4	Módulo TTGO T-Call V1.3 ESP32	9
2.4.1	Módulo SIM800L	10
2.4.2	Diagrama funcional do módulo SIM800L	12
2.4.3	Pinout do módulo SIM800L	12
2.5	Sensores para monitorização de parâmetros fisiológicos	13
2.5.1	Sensor MAX30100	13
2.5.2	Sensor MLX90614	15
2.6	Comunicação com sistemas online	18
2.6.1	Servidor na <i>Cloud</i>	18
2.6.2	Protocolo de comunicação MQTT	19
3	Desenvolvimento e implementação	21
3.1	Ligação dos dois módulos de aquisição de sinais	21
3.2	Fluxograma	22
3.2.1	Módulo do utente 1 (Wi-Fi)	23
3.2.2	Módulo do utente 2 (GPRS)	24
3.3	Servidor em nuvem utilizado no projeto	26
3.3.1	Mosquitto <i>broker</i>	27
3.3.2	Node-RED	27
3.3.3	Arquitetura do sistema	28
3.4	Programa em Node-RED	29
3.4.1	Flow saturação de oxigénio	29
3.4.2	Flow frequência cardíaca	30
3.4.3	Flow temperatura	31
3.4.4	Flow para o registo dos alertas na base de dados	32

3.4.5	Nós para representação das informações na UI	33
3.4.6	<i>Dashboard</i>	34
3.5	Resultados.....	35
4	Modelo de negócio	39
4.1	Custos.....	41
4.2	Projeto Piloto	44
5	Conclusões.....	45
5.1	Trabalhos futuros	45

Lista de Figuras

Figura 2 - Diagrama de blocos do ESP32 (ESP32 Series Datasheet Version 4.7 2.4 GHz Wi-Fi + Bluetooth [®] + Bluetooth LE SoC Including, n.d.)	8
Figura 3 - <i>Pinout</i> do módulo TTGO T-Call V1.3 ESP32(TTGO T-Call V1.3 ESP32 <i>Pinout</i> » DIY <i>Usthad</i> , n.d.).....	9
Figura 4 - Vista superior e inferior do módulo SIM800L (<i>Datasheet_SIM800L</i> , n.d.)	10
Figura 5 - Diagrama funcional do módulo SIM800L (<i>Datasheet_SIM800L</i> , n.d.).....	12
Figura 6 - <i>Pinout</i> do módulo SIM800L (<i>Sim800L GSM Module Pinout (as Adopted from... Download Scientific Diagram</i> , n.d.).....	12
Figura 7 - Diagrama dos pinos do módulo SIM800L (vista de cima)(<i>Datasheet_SIM800L</i> , n.d.)	13
Figura 8 - <i>Pinout</i> do sensor MAX30100(RCWL-0530/MAX30100 an <i>Oximetry / Heart Rate / Temperature Sensor – Peko’s Library</i> , n.d.)	14
Figura 9 - Esquemático representativo do sensor MAX30100(RCWL-0530/MAX30100 an <i>Oximetry / Heart Rate / Temperature Sensor – Peko’s Library</i> , n.d.)	15
Figura 12 - <i>Pinout</i> do sensor MLX90614 (<i>MLX90614 Contactless IR Temperature Sensor Interfacing with Arduino</i> , n.d.)	16
Figura 13 - Esquemático do sensor MLX90614 (MLX90614 Family Datasheet Single and Dual Zone Infra-Red Thermometer in TO-39 Features and Benefits, n.d.)	17
Figura 14 - Diagrama de blocos do sensor MLX90614(MLX90614 Family Datasheet Single and Dual Zone Infra Red Thermometer in TO-39 Features and Benefits, n.d.)	17
Figura 17 - Ilustração da arquitetura de um servidor em nuvem (<i>Arquitetura de Computação Em Nuvem Diagrama de Rede de Computadores, Aplicativo Web, Rede de Computadores, Texto, Serviço Png PNGWing</i> , n.d.).....	18
Figura 19 - Arquitetura do protocolo de comunicação MQTT (<i>MQTT – Wikipédia, a Enciclopédia Livre</i> , n.d.).....	20
Figura 20 - Esquema de ligação do módulo do utente 1	21
Figura 21 - Esquema de ligação do módulo do utente 2	22
Figura 22 - Fluxograma do funcionamento geral do sistema	23
Figura 23 - Fluxograma do funcionamento do módulo do Utente 1	24
Figura 24 - Fluxograma do funcionamento do módulo do Utente	25
Figura 26 - <i>Interface</i> do servidor MQTT criado para este projeto	26
Figura 27 - Linha de comandos do Linux Ubuntu VM	27
Figura 31 - Ilustração da arquitetura definida para o sistema.....	28
Figura 32 - <i>Flow</i> do processamento dos valores lidos de saturação de oxigénio	29
Figura 33 - <i>Flow</i> do processamento dos valores lidos de frequência cardíaca.....	30
Figura 35 - <i>Flow</i> do processamento dos valores lidos de temperatura corporal	31
Figura 36 - <i>Flow</i> do processamento do registo de alertas de cada Utente	32
Figura 37 - Nós relativos às informações dos utentes e às tabelas de valores ideais.....	33
Figura 38 – Vista geral do <i>dashboard</i> do utente e respetiva descrição das suas componentes	34
Figura 39 - Teste do módulo do Utente 1	35
Figura 40 - Visualização dos valores na <i>interface</i> de utilizador	36

Figura 41 - Alerta gerado por leitura critica	37
Figura 42 - Exemplo de e-mail enviado ao familiar do idoso/utente.....	37
Figura 43 - Alertas guardados em base de dados própria.....	38

Lista de Tabelas

Tabela 1 - Especificações chave do modulo SIM800L(<i>Datasheet_SIM800L</i> , n.d.).....	11
Tabela 8 - Tabela representativa do registo de configuração do Led.....	15
Tabela 12 - Custos do módulo do utente.....	41
Tabela 13 - Custos do servidor.....	42

Acrónimos e Símbolos

Lista de Acrónimos

ADC	<i>Analogic to Digital Converter</i>
API	<i>Application Programming Interface</i>
BPM	Batidas por minuto
GPRS	<i>General Packet Radio Service</i>
GSM	<i>Global System for Mobile Communications</i>
IoT	<i>Internet of Things</i>
LED	<i>Light Emitting Diode</i>

1 Introdução

O isolamento social dos idosos em Portugal é uma questão de saúde pública crescente, com implicações significativas para o bem-estar físico e mental desta faixa etária. Estudo após estudo aponta que uma parte considerável da população idosa em Portugal vive em condições de solidão, o que afeta diretamente a qualidade de vida e aumenta o risco de diversas doenças, como depressão, ansiedade e outras condições psicossociais. A literatura mostra que o envelhecimento da população portuguesa, aliado a fatores como a dispersão geográfica das famílias e a falta de redes de apoio, contribui para o aumento do isolamento social entre os idosos. Segundo um estudo realizado pelo Instituto Nacional de Estatística (INE) em 2019, aproximadamente 30% dos idosos em Portugal vivem sozinhos, e uma parte significativa deles reporta sentir-se isolada. Este fenómeno é particularmente prevalente entre as pessoas com mais de 80 anos, que muitas vezes enfrentam dificuldades de mobilidade, doenças crónicas e a perda de familiares próximos, o que agrava a sua situação social [1]. Além disso, o isolamento social entre os idosos não está apenas relacionado com a ausência de interações sociais, mas também com a falta de atividades que promovam a interação e o envolvimento com a comunidade, o que pode levar a um ciclo de solidão e depressão. Estudos sobre a saúde mental dos idosos também revelam a relação direta entre o isolamento social e o aumento do risco de doenças mentais, como a depressão. De acordo com a Organização Mundial de Saúde (OMS), a solidão pode ser um fator de risco importante para a saúde mental, e está associada a um aumento da mortalidade, doenças cardíacas e diminuição da qualidade de vida [2].

1.1 Soluções equivalentes já desenvolvidas

Este capítulo visa apresentar algumas das soluções já desenvolvidas, utilizando alguns dos métodos e tecnologias presentes neste projeto. Estas soluções apresentam alguns pontos fortes, no entanto são insuficientes num projeto de grande escala em termos geográficos e também, possuem um grau de complexidade de utilização, que pode não ser muito intuitivo para o utilizador final.

1.1.1 Solução desenvolvida por alunos do Instituto Politécnico de Leiria

Segundo [3] este sistema permite realizar a medição dos parâmetros biomédicos (Acelerómetro, Sensor de Temperatura e ECG), os dados medidos podem ser armazenados num cartão SD e enviados por Bluetooth para um *Smartphone*, onde podem ser visualizados numa aplicação desenvolvida para este efeito. É também possível emitir alarmes sonoros e luminosos caso os valores medidos sejam considerados anormais, e visualização dos mesmos num LCD.

Pontos Fortes:

- Portabilidade
- Armazenamento dos dados
- Visualização de dados no smartphone pessoal

Pontos Fracos:

- Não permite realização de medições a longa distância
- Grau de complexidade elevado para o utilizador final (Idoso)
- Necessário possuir um smartphone para visualização dos dados

1.1.2 Solução desenvolvida por alunos da Faculdade de Medicina da Universidade do Porto e da Faculdade de Ciências da Universidade do Porto.

Segundo [4] este projeto, baseia-se na criação de um kit, chamado Kit Mii, de monitorização de sinais vitais, que permite ao próprio idoso, recolher os seus dados de um modo simples e acessível, e que no fim desse processo seja possível o profissional de saúde ter conhecimento do estado de saúde do idoso, sem que este último se tenha de deslocar ao centro

de saúde ou vice-versa, através do envio desses dados para uma base de dados do centro de saúde. Clique ou toque aqui para introduzir texto.

Pontos Fortes:

- Monitorização á distancia
- Na compra do Kit tem a oferta de telefone para realizar chamadas para números de emergência.
- Sistema Revolucionário

Pontos Fracos:

- Necessidade de conhecimentos avançados por parte do idoso caso a aplicação falhe.
- Indisponibilidade do centro de saúde para analisar os dados.
- Requer ligação á Internet
- Nem todos os idosos poderão conseguir utilizar o sistema (baixa literacia digital, falta de memoria ou dificuldades de visão)

1.1.3 Solução desenvolvida por alunos e docente da Universidade SRM na India

Segundo [5] este projeto baseia-se num sistema remoto de monitorização de saúde controlado por Raspberry pi. O sistema é projetado para monitorizar continuamente os parâmetros vitais, como frequência cardíaca, pressão arterial e temperatura corporal. A informação é armazenada num servidor baseado em *Cloud* e pode ser consultada por meio de uma *interface* Web ou aplicativo móvel autorizado. O sistema envia também um alerta por GSM para o médico/família caso sejam detetadas irregularidades nos valores medidos.

Pontos Fortes:

- Preço
- Monitorização á distancia
- Informação disponíveis em plataforma Web
- Armazenamento em banco de dados
- Vasta gama de sensores

Pontos Fracos:

- Complexidade na utilização por parte do utilizador final (Idoso)

- Baixa portabilidade do sistema
- Dimensões do Hardware utilizado
- Possível indisponibilidade dos médicos

2 Bases Científicas e Tecnológicas

2.1 Monitorização de parâmetros fisiológicos

Segundo [7] “À medida que aumenta o interesse em melhorar a qualidade de vida, cresce também a necessidade de manter uma boa saúde no dia a dia. O monitoramento da saúde dos idosos é necessário para prevenir doenças e preservar o estilo de vida independente desse segmento em rápido crescimento da população global. Esses requisitos significam que os serviços médicos que antes eram restritos a hospitais e se concentravam na cura de doenças podem se estender diretamente às residências das pessoas para apoiar seus cuidados diários de saúde.” A análise regular destes parâmetros pode facilitar o diagnóstico eficaz de algumas patologias, como a taquicardia (elevada frequência cardíaca em repouso acima das 100 batidas por minuto) [8], braquicardia (baixa frequência cardíaca em repouso, abaixo das 60 batidas por minuto) [9] e hipoxia (níveis de oxigênio no sangue inferiores a 80 %) [9]

2.1.1 Frequência Cardíaca

Clique ou toque aqui para introduzir texto. A frequência cardíaca (batimento cardíaco) é o número de batidas do coração por minuto (BPM) e pode variar conforme as necessidades do corpo, como a absorção de oxigênio e a liberação de dióxido de carbono. É influenciada por fatores como genética, aptidão física, *stress*, dieta, consumo de medicamentos, hormônios, ambiente e doenças, além da interação entre esses elementos. [11]

2.1.2 Saturação de Oxigênio

Clique ou toque aqui para introduzir texto. A saturação de oxigênio mede a percentagem de oxihemoglobina (hemoglobina ligada ao oxigênio) no sangue, e é representada como saturação arterial de oxigênio (SaO₂) e saturação venosa de oxigênio (SvO₂). A saturação de oxigênio é um parâmetro vital para definir o conteúdo de oxigênio no

sangue e o fornecimento de mesmo [13]. Para a maioria dos adultos e crianças saudáveis, um nível normal de saturação de oxigênio está entre 95% e 100% [14]

2.1.3 Foto pletismografia

Clique ou toque aqui para introduzir texto. A forma de onda da foto pletismografia foi descrita pela primeira vez na década de 1930. Embora seja considerado um monitor auxiliar interessante, a “forma de onda de pulso” nunca foi submetida a uma investigação intensiva. A sua importância na medicina clínica aumentou muito com a introdução do oxímetro de pulso no atendimento clínico de rotina na década de 1980. A sua forma de onda agora é comumente exibida no ambiente clínico. Esforços de pesquisa ativa estão a começar a demonstrar uma utilidade além da saturação de oxigênio e determinação da frequência cardíaca. As tendências futuras estão sendo fortemente influenciadas pelo processamento de sinal digital moderno, que está a permitir um reexame dessa forma de onda onipresente. A chave para desbloquear o potencial dessa forma de onda é um acesso irrestrito ao sinal bruto, combinado com a padronização de sua apresentação e métodos de análise. A longo prazo, é necessário aprender como quantificar consistentemente as características da foto pletismografia de forma a permitir que os resultados dos esforços de pesquisa sejam traduzidos em dispositivos clinicamente úteis. A foto pletismografia trata-se, portanto de uma técnica ótica que pode ser usada na detecção das alterações no volume sanguíneo no leito microvascular do tecido. A foto pletismografia é frequentemente obtida usando um oxímetro de pulso ou dedo que utiliza LEDs vermelho e infravermelho a fim de medir as mudanças na absorção de luz. Um oxímetro de pulso convencional monitoriza a perfusão do sangue para a derme e tecido subcutâneo da pele. O coração bombeia sangue para a periferia em cada ciclo cardíaco. As ondas de pressão são amortecidas devido as reflexões que a onda principal (vinda do coração) faz ao se propagar pelas artérias torácicas, ílica e renal. Nesses sítios de ligação (ou mudanças de calibre das artérias) ocorrem as reflexões. Mesmo que este pulso de pressão seja amortecido ao chegar na ponta do dedo, já é suficiente para distender as artérias e arteríolas no tecido subcutâneo. O volume altera-se devido ao pulso de pressão, e é detetado a partir da iluminação da superfície da pele com uma luz de um díodo emissor de luz (LED) e, em seguida, mede-se a quantidade de luz transmitida ou refletida para um foto-díodo. Graficamente, cada ciclo cardíaco aparece como um pico. Como o fluxo sanguíneo para a pele pode ser modulado por diferentes sistemas fisiológicos, a foto pletismografia também pode ser usada para monitorizar a respiração, a hipovolemia e outras condições circulatórias. Além disso, o formato do pulso do PPG difere de pessoa para pessoa e varia de acordo com o posicionamento e a maneira em que o oxímetro está conectado [15]

2.1.4 Oximetria de pulso

A oximetria de pulso é um método utilizado para medir a quantidade de oxigênio que o sangue transporta através da hemoglobina. Usando um pequeno dispositivo chamado oxímetro de pulso, o nível de oxigênio sanguíneo pode ser aferido sem a necessidade de utilização de métodos invasivos. O nível de oxigênio medido com um oxímetro é chamado de nível de saturação de oxigênio (abreviado como O2sat ou SpO2). A SpO2 é a porcentagem de oxigênio que o sangue está a transportar, comparada com o máximo da sua capacidade de transporte. Idealmente, mais de 89 % dos glóbulos vermelhos devem atuar no transporte de oxigênio [12]

2.2 IoT – Internet das coisas

Clique ou toque aqui para introduzir texto. A proliferação de objetos inteligentes com capacidade de sensoriamento, processamento e comunicação tem aumentado nos últimos anos. Neste cenário, a Internet das Coisas conecta estes objetos à Internet e promove a comunicação entre utilizadores e dispositivos. A IoT possibilita uma grande quantidade de novas aplicações, as quais tanto a academia quanto a indústria podem beneficiar se, tais como cidades inteligentes, saúde e automação de ambientes. O objetivo (benigno) de todos estes dispositivos e sobretudo da grande quantidade de dados resultantes da respetiva interação através da Internet, é que o processamento resultante seja efetuado de forma que, por exemplo, se evitem engarrafamentos de trânsito, se antecipe atempadamente uma doença fatal num doente ou um incidente num edifício, se utilize de forma mais eficiente a energia, para dar apenas alguns exemplos [16]

2.2.1 IoT na Medicina

Além de melhorar o atendimento ao paciente, a IoT pode reduzir substancialmente os preços dos serviços de saúde ao simplificar processos, automatizar tarefas de rotina e diminuir a necessidade de intervenções caras. Ela melhora o atendimento ao paciente ao permitir a coleta e a monitorização dos dados em tempo real, otimizando assim os tempos de resposta e os resultados dos pacientes. As tecnologias de IoT simplificam as operações de saúde ao automatizar tarefas administrativas, o que não só economiza tempo, mas também reduz a necessidade de uma extensa força de trabalho administrativa, resultando em economia de custos. Além disso, tarefas de rotina como monitorização de sinais vitais e dispensação de medicamentos podem ser automatizadas, diminuindo o risco de erros e diminuindo os custos de mão de obra. Mais significativamente, os recursos de deteção precoce e monitorização contínua da IoT ajudam a evitar complicações médicas dispendiosas, reduzindo hospitalizações e mitigando a necessidade de intervenções caras, reduzindo significativamente os custos de saúde [17]

2.3 Sensor PPG baseado em ESP32

Este projeto está alinhado com o paradigma do IoT, os seus componentes e o seu sistema por inteiro. Foi então decidido para efeitos de execução do mesmo, que a plataforma utilizada seria o processador ESP32, com o seu poder de computação (32 bit's Dual-Core) necessário para o processamento de sinal, a sua *interface* Wi-Fi combinada com um módulo SIM800L que permite efetuar uma ligação á Internet em localizações remotas (onde o sinal Wi-Fi é mais fraco) através da rede 2G que utiliza a tecnologia GPRS[18] Criado pela Espressif Systems, o ESP32 é um sistema de baixo custo e baixo consumo de energia da série SoC com recursos de Wi-Fi e Bluetooth de modo duplo. A família ESP32 inclui os chips ESP32-D0WDQ6 (e ESP32-D0WD), ESP32-D2WD, ESP32-S0WD e o pacote de sistema (SiP) ESP32-PICO-D4. No seu núcleo, existe um microprocessador Tensilica Xtensa LX6 dual-core com uma frequência de até 240 MHz. O ESP32 é altamente integrado com interruptores de antena embutidos, RF balun, amplificador de potência, amplificador de receção de baixo ruído, filtros e módulos de gerenciamento de energia. Projetado para dispositivos móveis, eletrônicos vestíveis e aplicativos de IoT, o ESP32 atinge um consumo de energia ultrabaixo por meio de recursos de economia de energia, como os modos modem-sleep, light-sleep e deep-sleep. (a corrente consumida pode baixar até 20 μ A). Cada um destes modos desativa partes do sistema em ordem para poupar energia [19]

Na figura 1 podemos verificar o respetivo diagrama de blocos do ESP32.

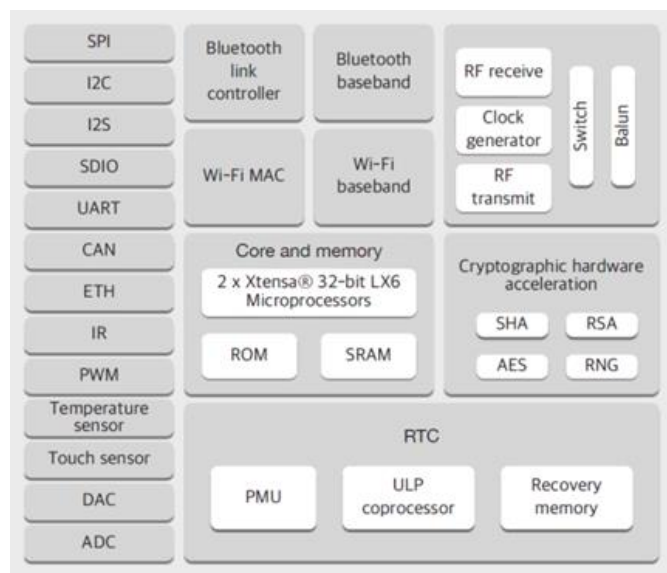


Figura 1 - Diagrama de blocos do ESP32 [19]

2.4 Modulo TTGO T-Call V1.3 ESP32

Para transmitir e receber dados, os dispositivos IoT dependem de uma ligação de rede. A perda desta ligação implica a perda das funcionalidades do dispositivo. Embora existam diversas opções de conectividade para IoT, cada uma delas é mais indicada para diferentes tipos de cobertura. A escolha da solução pode restringir consideravelmente as opções de implantação, tornando a cobertura um desafio constante em projetos de IoT. Um exemplo comum é o uso da tecnologia Wi-Fi para conectividade. No entanto, os dispositivos só funcionam dentro de uma área limitada, próximos de um *router*, o que restringe a implementação a locais com cobertura Wi-Fi. Caso esta infraestrutura não esteja disponível, será necessário investir na sua criação ou equipar os dispositivos com uma solução alternativa que já possua cobertura. A tecnologia móvel é a solução mais popular e flexível para a comunicação de dados em projetos IoT. As suas diversas tecnologias de transmissão, as redes 2G, 3G, 4G e mais recentemente o 5G, proporcionam ampla cobertura, permitindo que os dispositivos IoT funcionem a vários quilómetros de distância de uma infraestrutura de rede [20]

Tendo em conta que este projeto foi pensado para ser aplicado maioritariamente em aldeias isoladas ou em localizações remotas, onde muitas vezes não existe uma infraestrutura Wi-Fi, esta seria então uma boa solução para conseguirmos obter uma ligação à Internet, mantendo assim intactas, todas as funcionalidades deste sistema. Como já mencionado, para a realização deste projeto foi então utilizada a plataforma ESP32 com o módulo SIM800L integrado. Na figura 2 podemos verificar o *pinout* do módulo TTGO T-Call V1.3 ESP32:

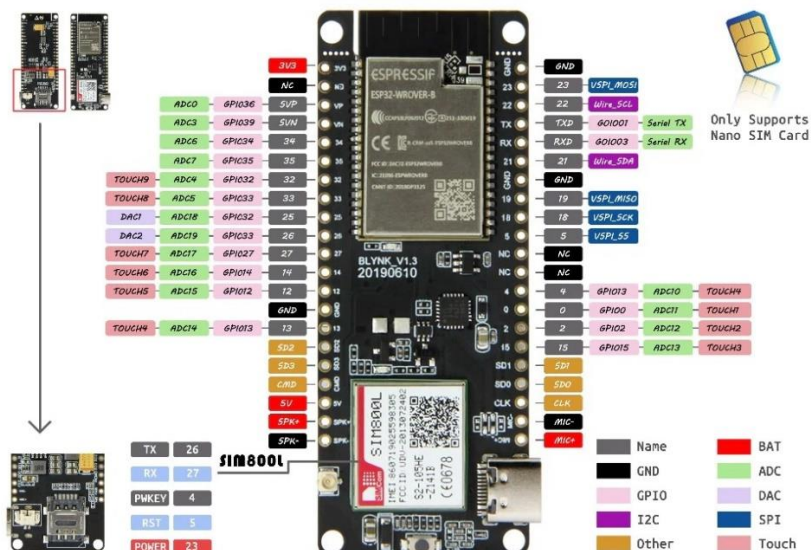


Figura 2 - Pinout do módulo TTGO T-Call V1.3 ESP32[21]

2.4.1 Módulo SIM800L

O ESP32 TTGO T-Call possui um módulo SIM800L integrado, este módulo permite a ligação á rede 2G através da tecnologia GPRS, permite também o envio e receção de SMS e de chamadas de voz. É um componente de baixo custo, de pequenas dimensões e que suporta a frequência de banda quádrupla. Tudo isto faz deste componente uma solução ajustada para qualquer projeto que requeira conectividade à Internet em qualquer ponto do mundo [22], [23] a figura 3 ilustra o chip SIM800L, integrado no modulo ESP32 TTGO T-Call.



Figura 3 - Vista superior e inferior do módulo SIM800L [24]

Na seguinte tabela, podemos verificar as principais características e funcionalidades do modulo SIM800L:

Tabela 1 - Especificações chave do modulo SIM800L[24]

Feature	Implementation
Power supply	3.4V ~4.4V
Power saving	typical power consumption in sleep mode is 0.7mA (AT+CFUN=0)
Frequency bands	<ul style="list-style-type: none"> ● Quad-band: GSM 850, EGSM 900, DCS 1800, PCS 1900. SIM800L can search the 4 frequency bands automatically. The frequency bands can also be set by AT command "AT+CBAND". For details, please refer to document [1]. ● Compliant to GSM Phase 2/2+
Transmitting power	<ul style="list-style-type: none"> ● Class 4 (2W) at GSM 850 and EGSM 900 ● Class 1 (1W) at DCS 1800 and PCS 1900
GPRS connectivity	<ul style="list-style-type: none"> ● GPRS multi-slot class 12 (default) ● GPRS multi-slot class 1~12 (option)
Temperature range	<ul style="list-style-type: none"> ● Normal operation: -40°C ~ +85°C
Data GPRS	<ul style="list-style-type: none"> ● GPRS data downlink transfer: max. 85.6 kbps ● GPRS data uplink transfer: max. 85.6 kbps ● Coding scheme: CS-1, CS-2, CS-3 and CS-4 ● PAP protocol for PPP connect ● Integrate the TCP/IP protocol. ● Support Packet Broadcast Control Channel (PBCCH) ● CSD transmission rates: 2.4, 4.8, 9.6, 14.4 kbps
CSD	<ul style="list-style-type: none"> ● Support CSD transmission
USSD	<ul style="list-style-type: none"> ● Unstructured Supplementary Services Data (USSD) support
SMS	<ul style="list-style-type: none"> ● MT, MO, CB, Text and PDU mode ● SMS storage: SIM card
SIM interface	Support SIM card: 1.8V, 3V
External antenna	Antenna pad
Audio features	<p>Speech codec modes:</p> <ul style="list-style-type: none"> ● Half Rate (ETS 06.20) ● Full Rate (ETS 06.10) ● Enhanced Full Rate (ETS 06.50 / 06.60 / 06.80) ● Adaptive multi rate (AMR) ● Echo Cancellation ● Noise Suppression
Serial port and debug port	<p>Serial port:</p> <ul style="list-style-type: none"> ● Full modem interface with status and control lines, unbalanced, asynchronous. ● 1200bps to 115200bps. ● Can be used for AT commands or data stream. ● Support RTS/CTS hardware handshake and software ON/OFF flow control. ● Multiplex ability according to GSM 07.10 Multiplexer Protocol. ● Autobauding supports baud rate from 1200 bps to 57600bps. ● upgrading firmware <p>Debug port:</p> <ul style="list-style-type: none"> ● USB_DM and USB_DP ● Can be used for debugging and upgrading firmware.
Phonebook management	Support phonebook types: SM, FD, LD, RC, ON, MC.
SIM application toolkit	GSM 11.14 Release 99
Real time clock	Support RTC
Timing functions	Use AT command set
Physical characteristics	Size:15.8*17.8*2.4mm Weight:1.35g
Firmware upgrade	Main serial port or USB port.

2.4.2 Diagrama funcional do módulo SIM800L

Na figura 4, podemos verificar os diferentes blocos de funcionamento do módulo SIM800L.

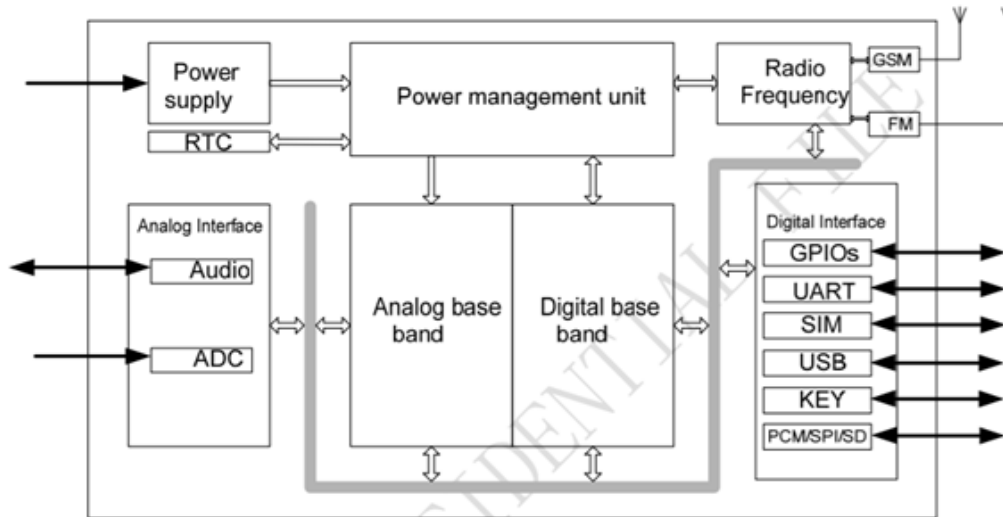


Figura 4 - Diagrama funcional do módulo SIM800L [24]

2.4.3 Pinout do módulo SIM800L

Na figura 5, podemos verificar o *pinout* do módulo SIM800L.



Figura 5 - Pinout do módulo SIM800L [25]

Na figura 6, podemos visualizar o diagrama de pinos do módulo SIM800L, onde é especificado a função de cada um deles:

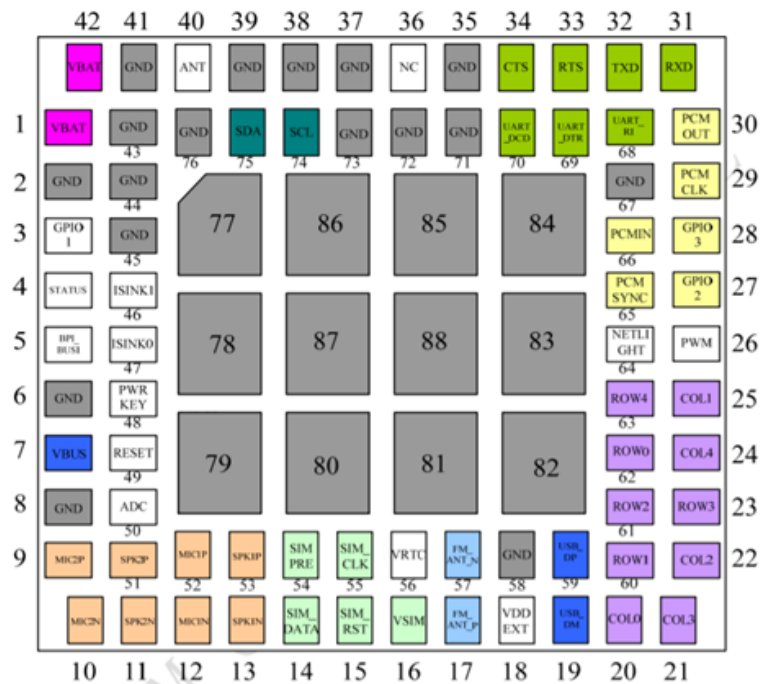


Figura 6 - Diagrama dos pinos do módulo SIM800L (vista de cima)[24]

2.5 Sensores para monitorização de parâmetros fisiológicos

De seguida são apresentados os sensores utilizados neste projeto. Os principais parâmetros fisiológicos a serem medidos foram a frequência cardíaca, saturação de oxigénio no sangue e temperatura corporal. Para isto, foram utilizados os sensores MAX30100 e o sensor MLX90614 respetivamente.

2.5.1 Sensor MAX30100

O MAX30100 é uma solução completa de oximetria de pulso e sensor de frequência cardíaca projetada para os exigentes requisitos de dispositivos usáveis. Ele consegue reunir num único chip o tamanho reduzido e um bom desempenho ótico e elétrico. Necessita do mínimo de hardware externo para ser integrado num dispositivo usável. O MAX30100 é totalmente configurável por meio de registos de software, e os dados de saída digital são armazenados num *buffer* FIFO de 16 bits dentro do dispositivo. O FIFO permite que o MAX30100 seja conectado a um microcontrolador ou microprocessador num barramento compartilhado, onde os dados não são lidos continuamente a partir dos registos do dispositivo [26]

Na figura 7 seguinte, pode-se verificar o *pinout* do sensor MAX30100:

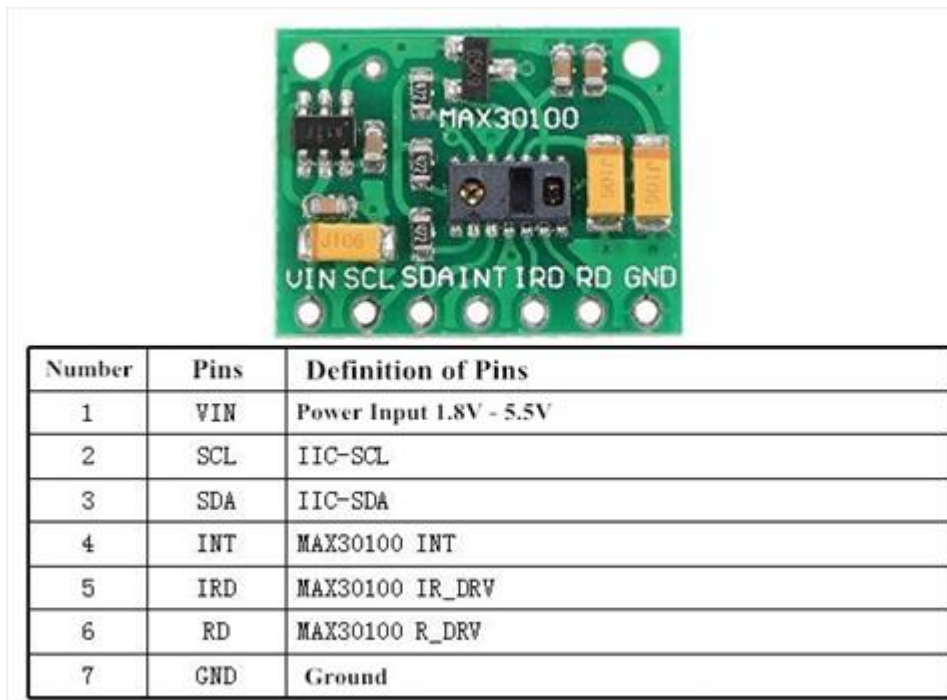


Figura 7 - Pinout do sensor MAX30100[27]

2.5.1.1 Características do sensor MAX30100 [26]

- LEDs integrados, foto sensor e *interface* analógica de alto desempenho.
- 14 pinos minúsculos (5,6 mm x 2,8 mm x 1,2 mm) sistema encapsulado melhorado opticamente.
- Taxa de amostragem e corrente de LED programáveis para economizar de energia.
- Corrente de desligamento ultrabaixa (0,7 μ A, tipicamente)
- Alto SNR fornece resiliência robusta do artefacto de movimento
- Capacidade de cancelamento da luz ambiente integrada
- Capacidade de alta taxa de amostragem
- Capacidade de saída rápida de dados

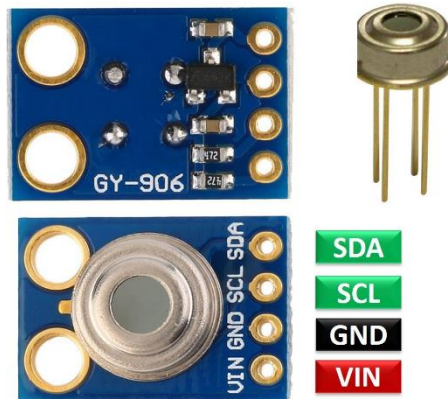


Figura 9 - Pinout do sensor MLX90614 [29]

2.5.2.1 Características do sensor MLX90614[28]

- Tamanho pequeno e baixo custo
- Fácil de integrar
- Calibrado de fábrica com ampla gama de temperatura: $-40\text{ }^{\circ}\text{C}$ a $+125\text{ }^{\circ}\text{C}$ para temperatura do sensor e de $-70\text{ }^{\circ}\text{C}$ a $+380\text{ }^{\circ}\text{C}$ para a temperatura do objeto.
- Alta precisão de $0,5\text{ }^{\circ}\text{C}$ numa faixa de temperatura alta ($0\text{...} +50\text{ }^{\circ}\text{C}$ para T_a e T_o)
- Calibração de alta precisão (médica) resolução de medição de $0,02\text{ }^{\circ}\text{C}$
- Versões de zona única e dupla
- *Interface* digital compatível com SMBus
- Saída PWM personalizável para leitura contínua
- Disponível nas versões 3V e 5V
- Adaptação simples para aplicações de $8\text{...} 16\text{V}$
- Modo de espera para redução do consumo de energia
- Diferentes opções de pacote para aplicativos e versatilidade de medidas
- Classe automotiva

Na figura 10, podemos verificar o esquema de ligação do sensor MLX90614.

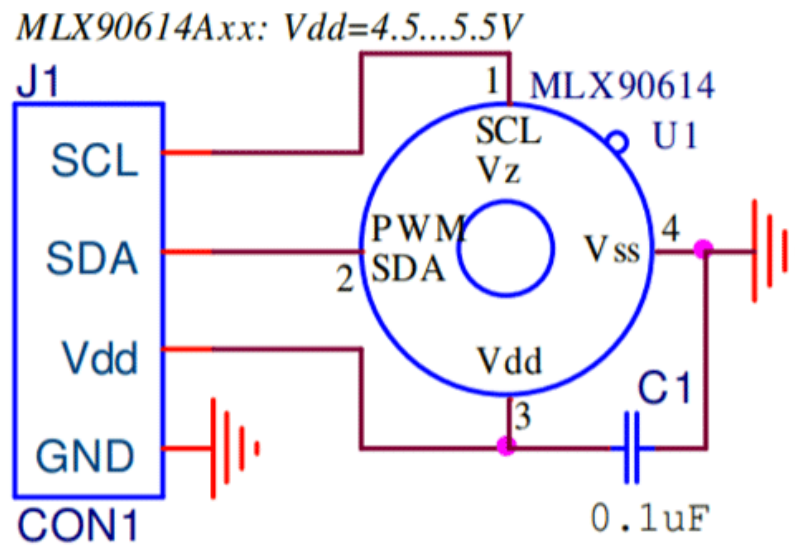


Figura 10 - Esquemático do sensor MLX90614 [28]

2.5.2.2 Processamento de sinal

A operação do MLX90614 é controlada por uma máquina de estados interna, que controla as medições e os cálculos da temperatura do objeto e do ambiente, e faz o pós-processamento destas temperaturas para emití-las por meio da saída PWM ou da *interface* compatível com SMBus. Este princípio de funcionamento, encontra-se ilustrado na figura 11 abaixo.

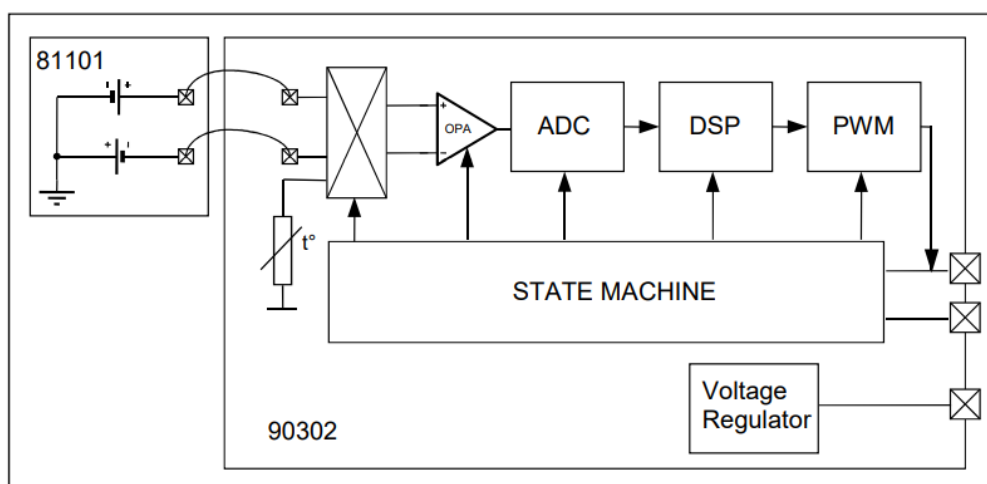


Figura 11 - Diagrama de blocos do sensor MLX90614[28]

2.6 Comunicação com sistemas online

A comunicação com sistemas online desempenha um papel crucial em projetos de IoT, pois permite a interligação e o fluxo de dados entre dispositivos físicos e plataformas digitais. Num cenário em que sensores, atuadores e dispositivos inteligentes recolhem e trocam informações em tempo real, a capacidade de comunicar com sistemas online, como servidores e bases de dados na nuvem, é essencial para a análise, monitorização e controlo remotos. Esta comunicação não só aumenta a eficiência operacional, mas também possibilita a implementação de soluções inteligentes, como a otimização de processos, a tomada de decisões automatizada e a resposta em tempo real a eventos, tornando o IoT uma tecnologia poderosa e transformadora.

2.6.1 Servidor na Cloud

Um servidor na *cloud* é uma infraestrutura de computação virtualizada que permite armazenar, gerir e processar dados através da internet, em vez de um servidor físico local. Ele oferece flexibilidade, escalabilidade e acessibilidade, permitindo que os utilizadores acedam aos recursos e aplicações em qualquer lugar e a qualquer momento. É ideal para empresas que procuram reduzir os custos operacionais e aumentar a eficiência. Estes servidores são utilizados para diversas finalidades, desde a hospedagem de websites até ao processamento de grandes volumes de dados [30]. A figura 12, demonstra um exemplo da arquitetura de um servidor em nuvem.

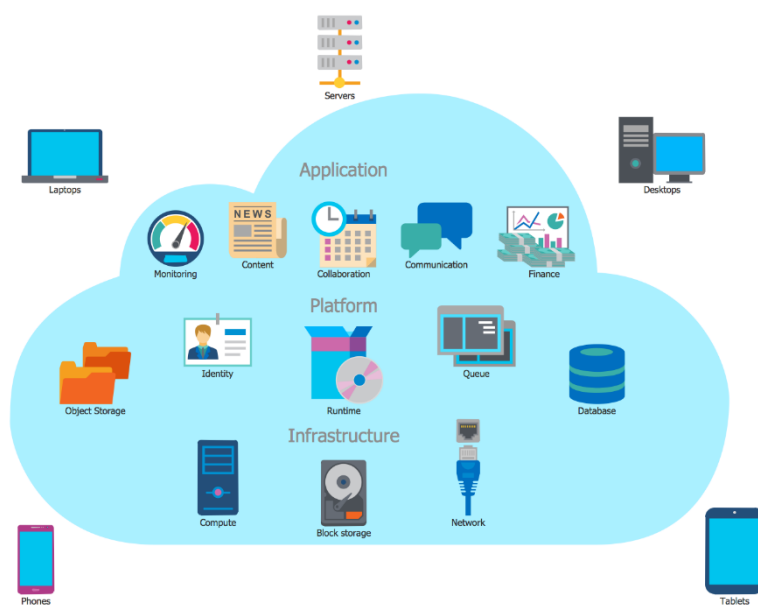


Figura 12 - Ilustração da arquitetura de um servidor em nuvem [31]

Elementos-chave

- Infraestrutura de computação que pode ser física (bare metal), virtual ou uma mistura das duas, dependendo do caso de uso.
- Tem todos os recursos de um servidor local.
- Permite que os utilizadores processem cargas de trabalho intensivas e que armazenem grandes volumes de informações.
- Serviços automatizados são acedidos sob demanda por meio de uma API.
- Oferece aos utilizadores a opção de pagamento mensal ou conforme o uso.
- Os utilizadores podem optar por um plano de hospedagem compartilhada que escala de acordo com as necessidades.

2.6.2 Protocolo de comunicação MQTT

Segundo [33] o MQTT (Message Queuing Telemetry Transport) é um protocolo de comunicação leve e baseado em mensagens, desenvolvido para ser utilizado em ambientes com recursos limitados, como dispositivos IoT. A sua principal característica é a capacidade de operar em redes com largura de banda reduzida, alta latência ou desconexões temporárias. O MQTT utiliza um modelo de publicação/assinatura, onde os dispositivos (clientes) podem inscrever-se em tópicos específicos e publicar mensagens sobre esses tópicos, de forma assíncrona. O protocolo foi inicialmente desenvolvido pela IBM nos anos 1990 para ser usado em sistemas de comunicação em ambientes de sensores remotos, e desde então tem-se consolidado como um dos padrões mais utilizados em projetos de IoT devido à sua eficiência e simplicidade. O funcionamento do MQTT pode ser compreendido através dos três componentes principais do protocolo: o *broker*, o *publisher* e o *subscriber*. O *broker* é o intermediário responsável por gerir a comunicação entre os dispositivos, recebendo mensagens publicadas e retransmitindo-as para os assinantes dos tópicos correspondentes. Os dispositivos podem atuar como *publishers*, que publicam mensagens sobre um determinado tópico, ou como *subscribers*, que se inscrevem para receber mensagens sobre tópicos específicos. Na figura 13 abaixo podemos verificar um diagrama ilustrativo destes três componentes.

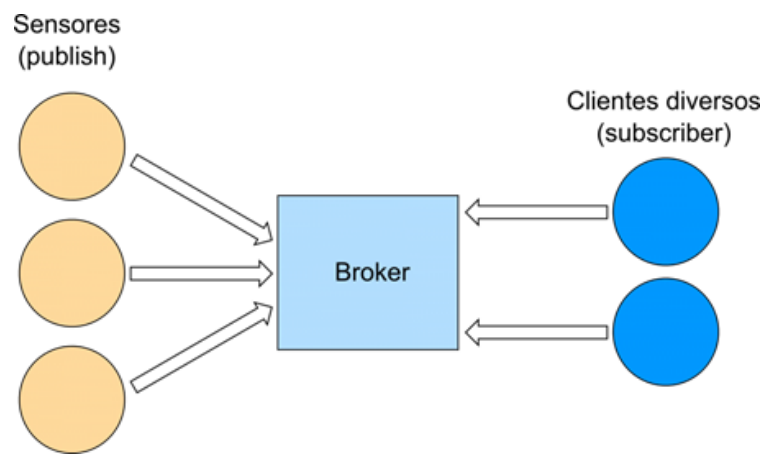


Figura 13 - Arquitetura do protocolo de comunicação MQTT [34]

3 Desenvolvimento e implementação

Este capítulo apresenta de forma detalhada o desenvolvimento e a implementação do sistema, abordando diversos aspetos fundamentais para a compreensão do seu funcionamento. Inicialmente, é apresentado o esquema de ligação dos módulos que compõem o sistema, em seguida, o fluxograma do princípio de funcionamento do sistema é exibido, fornecendo uma visão clara e objetiva dos processos envolvidos na execução das tarefas. Além disso, são destacadas as tecnologias e os softwares de programação utilizados, elucidando as ferramentas que possibilitaram a construção e o controlo do sistema. Por fim, são apresentados os resultados obtidos, fornecendo uma análise dos desempenhos alcançados.

3.1 Ligação dos dois módulos de aquisição de sinais

Nas figuras 14 e 15 seguintes, temos o esquema de ligação dos módulos de utente 1 e 2 respetivamente.

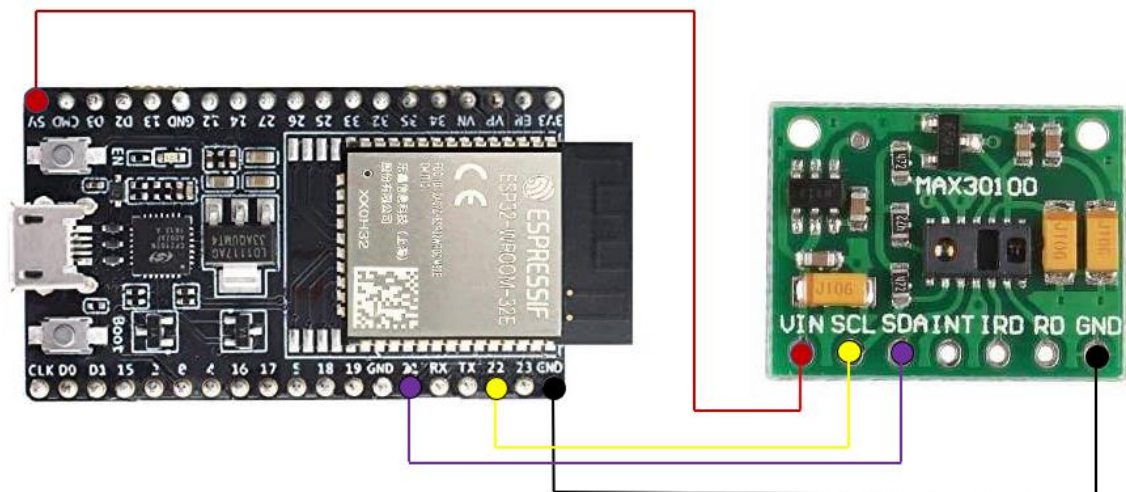


Figura 14 - Esquema de ligação do módulo do utente 1

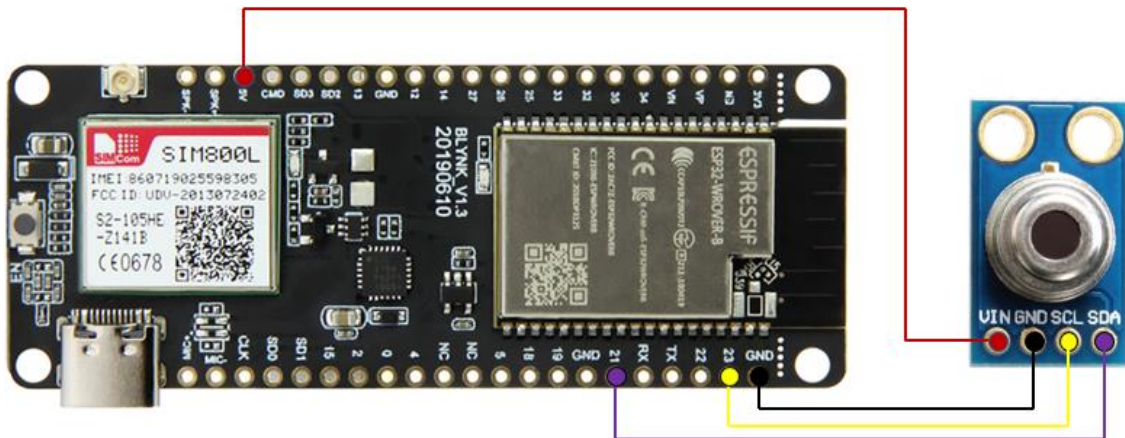


Figura 15 - Esquema de ligação do módulo do utente 2

A comunicação entre estes dispositivos é feita por I2C, que se trata de um protocolo de comunicação serie, half-duplex, desenvolvido pela Philips, em que os dados são transferidos bit a bit ao longo de um único fio (a linha SDA). Clique ou toque aqui para introduzir texto. O protocolo de comunicação I2C é síncrono, deste modo, a saída dos bits é sincronizada com a amostragem dos bits por um sinal de *clock* compartilhado entre o master e o slave. O sinal de clock é sempre controlado pelo master. Neste projeto o ESP32 funciona como master e o sensor MAX30100 como slave [35].

3.2 Fluxograma

Na figura 16 temos o fluxograma que demonstra o modo de funcionamento geral do sistema. O sistema é essencialmente composto por 2 blocos principais, o modulo de sensores e o servidor MQTT. Os dados são lidos pelo modulo de sensores e enviados para o servidor central através de uma ligação por Wi-fi ou GPRS. O servidor é então responsável pela visualização destes parâmetros e respetivo tratamento dos mesmo em casos de leituras críticas.

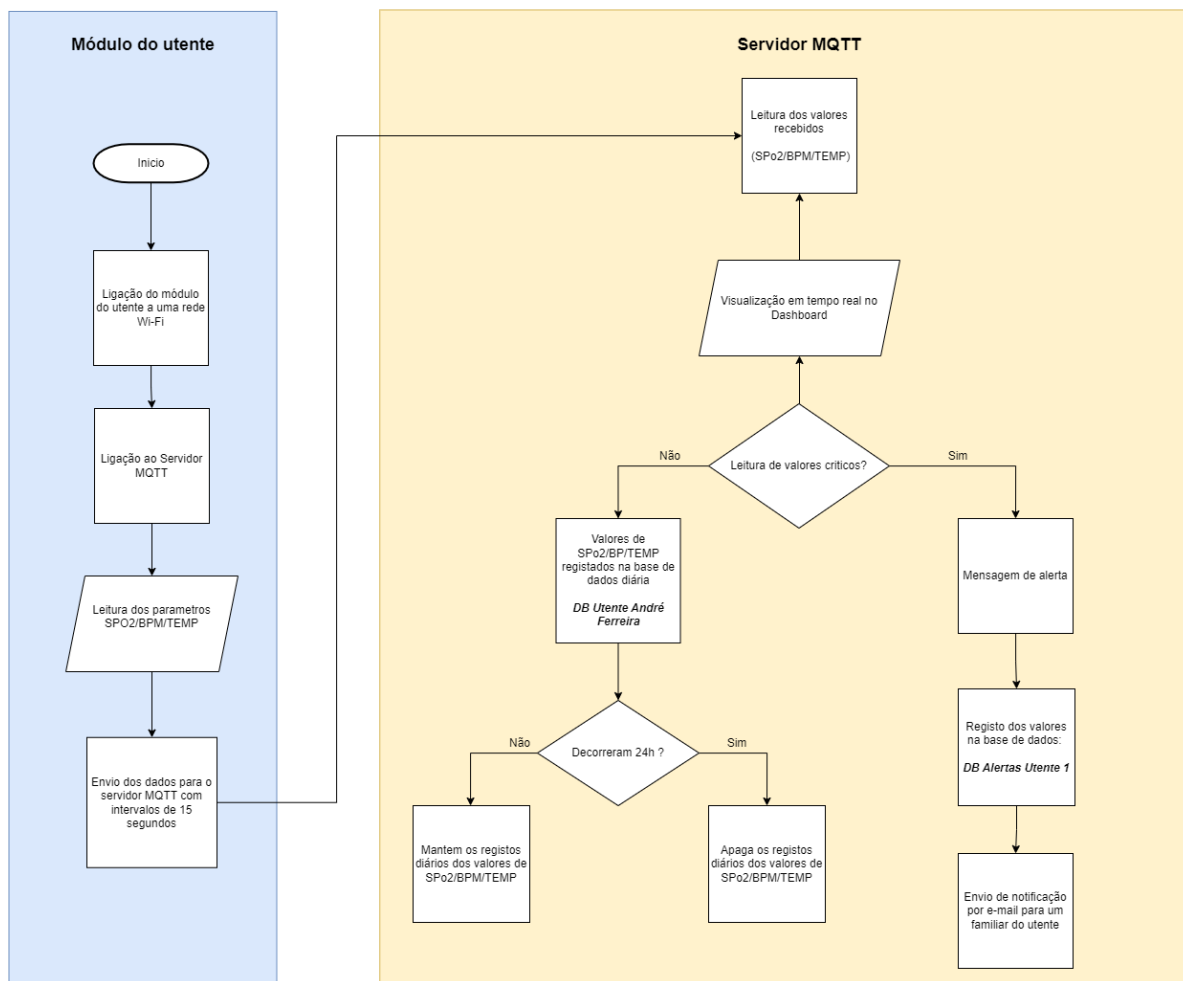


Figura 16 - Fluxograma do funcionamento geral do sistema

3.2.1 Módulo do utente 1 (Wi-Fi)

A figura 17, demonstra o modo de funcionamento do modulo do utente 1 composto com o modulo de sensores de frequência cardíaca e saturação de oxigénio MAX30100, que utiliza a tecnologia Wi-Fi para realizar a ligação ao servidor.

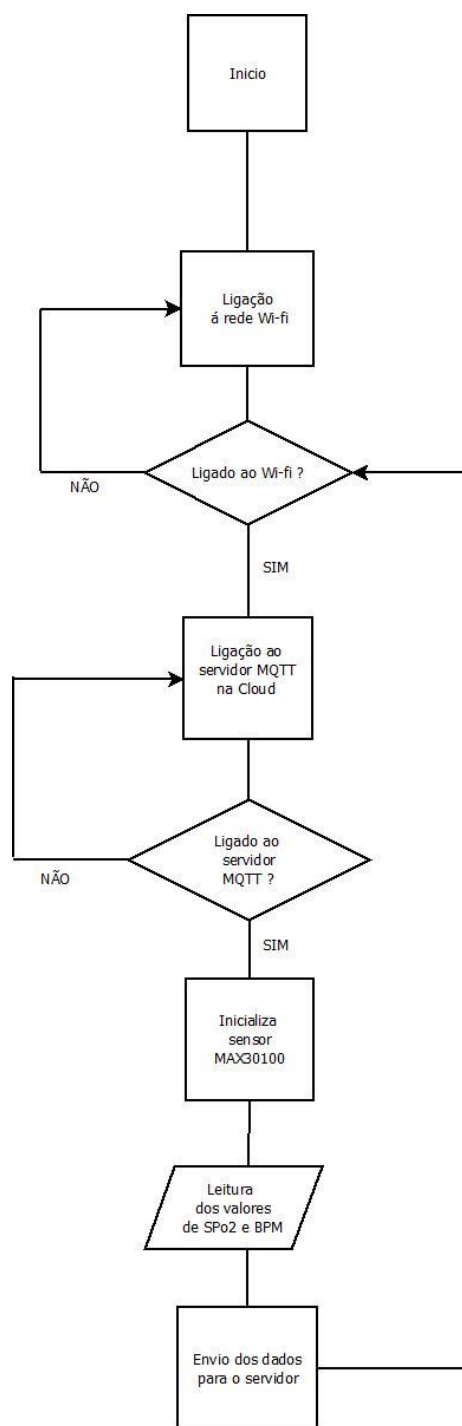


Figura 17 - Fluxograma do funcionamento do módulo do Utente 1

3.2.2 Módulo do utente 2 (GPRS)

A figura 18, demonstra o modo de funcionamento do modulo do utente 2 composto com o sensor de temperatura MLX90614, e que utiliza a tecnologia GPRS para realizar a ligação ao servidor.

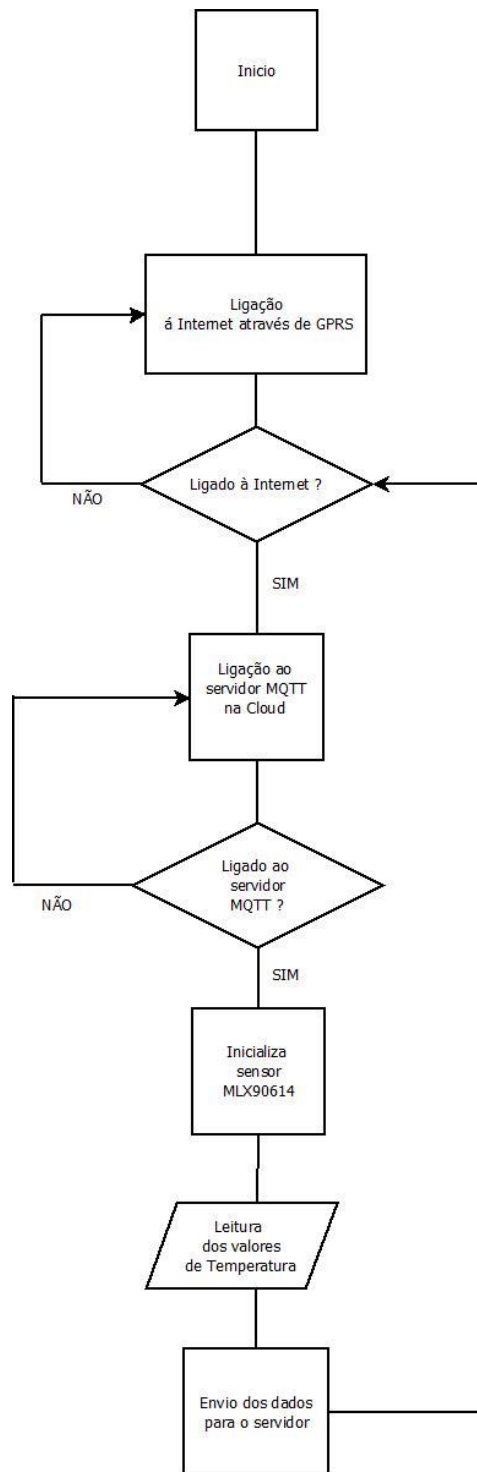


Figura 18 - Fluxograma do funcionamento do módulo do Utente

3.3 Servidor em nuvem utilizado no projeto

Para a execução deste projeto foi utilizado o provedor de serviços em nuvem DigitalOcean Inc. A escolha deste provedor recaiu sobre o a possibilidade de usufruir de um período de utilização grátis apenas para testar o funcionamento do sistema. A DigitalOcean fornece aos desenvolvedores serviços em nuvem que ajudam a implementar e a dimensionar aplicativos executados simultaneamente em vários computadores. Na figura 19 abaixo, temos um *screenshot* que demonstra um exemplo do ambiente de trabalho deste servidor utilizado no projeto.

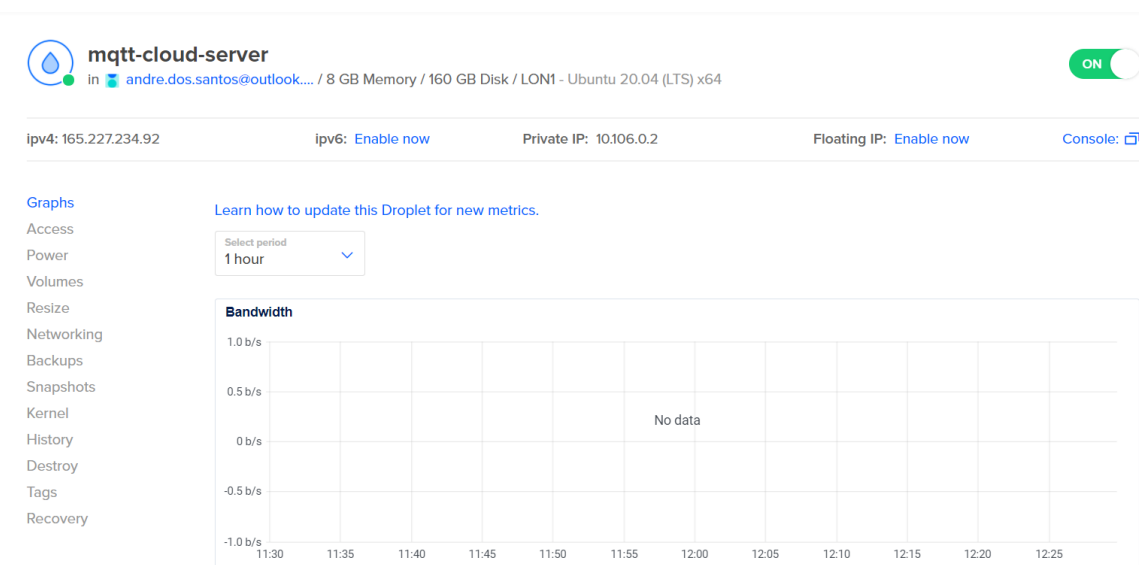
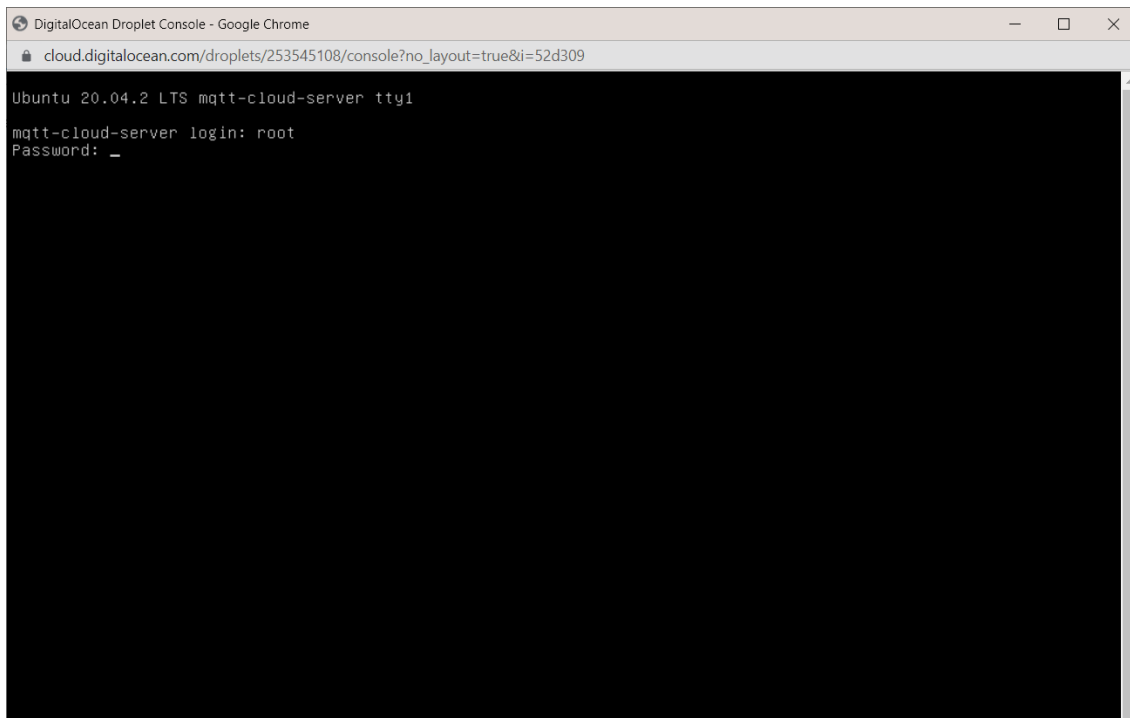


Figura 19 - Interface do servidor MQTT criado para este projeto

Dentro deste servidor podemos ter uma máquina virtual com o sistema operativo Linux Ubuntu, com acesso total ao terminal de comandos, o que permite instalar qualquer software adicional que seja necessário para a implementação do projeto. Este terminal (Figura 20) é protegido por nome de utilizador e palavra-passe (definidos pelo utilizador aquando da criação do servidor).



```
Ubuntu 20.04.2 LTS mqtt-cloud-server tty1
mqtt-cloud-server login: root
Password: _
```

Figura 20 - Linha de comandos do Linux Ubuntu VM

3.3.1 Mosquitto *broker*

Como mencionado anteriormente, o protocolo de comunicação MQTT contém um *broker* que permite gerir as publicações e as subscrições do protocolo, funcionando como um mediador da comunicação entre os diferentes dispositivos ligados ao servidor [36].

Para a criação de um *broker* MQTT foi instalado o software Mosquitto *Broker*. O Mosquitto é um agente de mensagens de *software* livre (licenciado por EPL / EDL) que implementa o protocolo MQTT nas versões 5.0, 3.1.1 e 3.1 [37]

O Mosquitto é leve e adequado para uso em todos os dispositivos, desde computadores de placa única de baixa potência até servidores completos. Como já mencionado anteriormente, o protocolo MQTT fornece um método leve de execução de mensagens usando um modelo de publicação / assinatura. Isso torna-o adequado para o uso em projetos de Internet das Coisas, onde são incorporados sensores de baixa potência ou dispositivos móveis, como telefones, computadores integrados ou microcontroladores [38].

3.3.2 Node-RED

O segmento da Internet das Coisas continua a crescer, assim como as plataformas de suporte à programação e visualização de dados. O Node-RED, foi criado pela IBM Emerging

Technology, e é uma ferramenta de desenvolvimento open-source. Esta plataforma, baseada em fluxos, foi desenvolvida para interagir com dispositivos de hardware, APIs e serviços online com vista a simplificar os sistemas IoT. O Node-RED permite a programação através de uma *interface* gráfica bastante simples e intuitiva. Para desenvolver uma aplicação com recurso ao Node-RED, o programador apenas tem de ligar nós uns aos outros, em que cada um deles tem a suas próprias funções. Esta plataforma permite também a escrita de código mais elaborado utilizando JavaScript. O Node-RED vem com um editor de fluxo, disponível via browser, que contem vários nodes (nós) por padrão que foram criados em JavaScript e NodeJS. Desta forma é possível criar aplicações avançadas de forma simples e intuitiva. Todo o código pode e deve ser dividido em *Flows* para uma melhor visualização. No final, esses *Flows* são armazenados em JSON simplificando assim a partilha do código [39]

3.3.3 Arquitetura do sistema

O *Broker* Mosquitto está em execução na nuvem (o serviço de *host* é fornecido pela Digital Ocean). Ele pode, desta forma, receber mensagens de diferentes módulos espalhados pela aldeia, permitindo que possam existir vários módulos ligados a redes diferentes, desde que estejam ligados ao *broker* MQTT na *Cloud*.

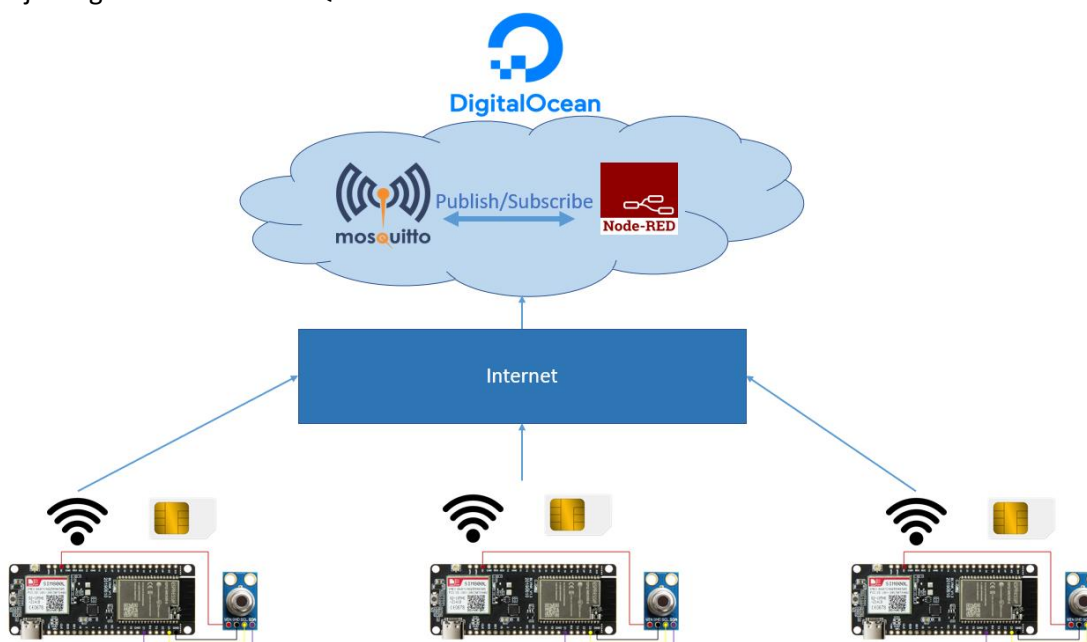


Figura 21 - Ilustração da arquitetura definida para o sistema

Cada módulo pode realizar uma ligação à Internet por Wi-Fi ou por GPRS (utilizando a rede 2G). Os dados recebidos destes módulos são processados pelo programa desenvolvido no *software* Node-RED, que é responsável pela visualização dos dados em tempo real (através da *User*

Interface) e armazenamento dos mesmos nas respetivas bases de dados. É também responsável pela emissão de alertas em caso de leituras de valores críticos, com notificação por e-mail para o familiar mais próximo do idoso, e registo desses mesmos alertas numa base de dados, de forma a manter um histórico de cada idoso.

3.4 Programa em Node-RED

Neste capítulo, temos os diferentes *flows* que gerem a leitura e o processamento dos parâmetros fisiológicos da frequência cardíaca, saturação de oxigénio e temperatura corporal.

3.4.1 Flow saturação de oxigénio

Na imagem seguinte, podemos verificar a programação da leitura e tratamento dos dados da saturação de oxigénio do utente 1 (André Ferreira).

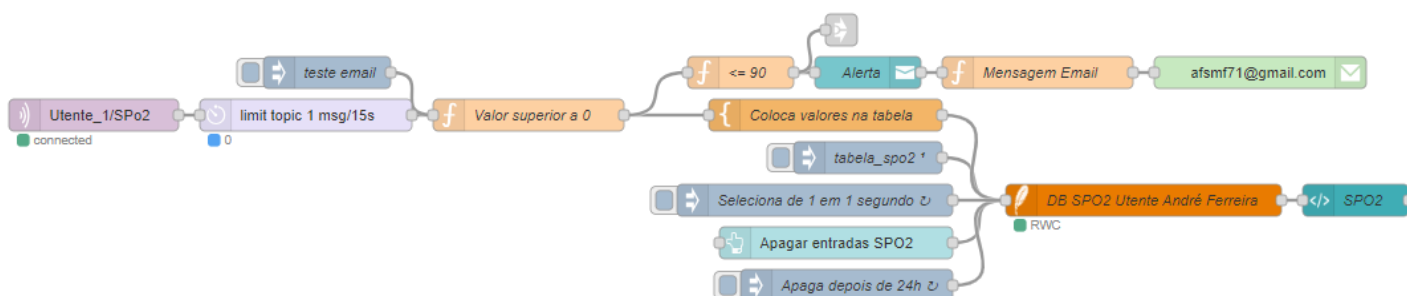


Figura 22 - Flow do processamento dos valores lidos de saturação de oxigénio

- As leituras são realizadas em intervalos de 15 segundos, e deverão ser superiores a 0 % de forma a validar as entradas, após isto os valores serão visualizados em tempo real no *Dashboard* do utilizador e colocados na base de dados dedicada, neste caso, será a *DB SPO2 Utente André Ferreira*.
- Caso os valores de oxigénio sejam menores ou iguais a 90, é gerado um alerta e é enviado a respetiva notificação para o e-mail do familiar do utente.
- Todas as medições estarão disponíveis durante 24 horas, no final deste período, estas medições serão apagadas de forma permanente.

3.4.2 Flow frequência cardíaca

Na imagem seguinte, podemos verificar a programação da leitura e tratamento dos dados da frequência cardíaca do utente 1 (André Ferreira).

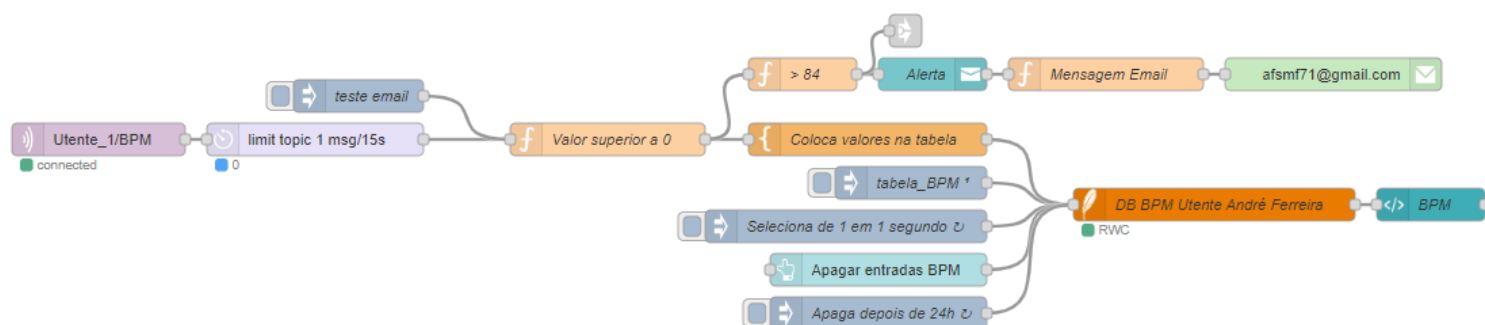


Figura 23 - Flow do processamento dos valores lidos de frequência cardíaca

- As leituras são realizadas em intervalos de 15 segundos, e deverão ser superiores a 0 BPM de forma a validar as entradas, após isto os valores serão visualizados em tempo real no *Dashboard* do utilizador e colocados na base de dados dedicada, neste caso, será a *DB BPM Utente André Ferreira*.
- Caso a frequência cardíaca em repouso for superior a 84, é gerado um alerta e é enviado a respetiva notificação para o e-mail do familiar do utente.
- Todas as medições estarão disponíveis durante 24 horas, no final deste período, estas medições serão apagadas de forma permanente.

3.4.3 Flow temperatura

Na imagem seguinte, podemos verificar a programação da leitura e tratamento dos dados da temperatura corporal do utente 1 (André Ferreira).

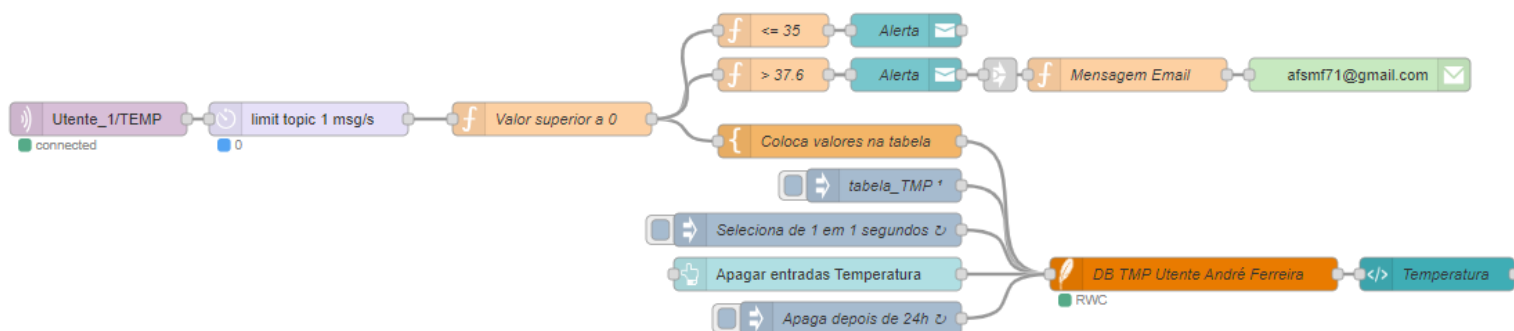


Figura 24 - Flow do processamento dos valores lidos de temperatura corporal

- As leituras são realizadas em intervalos de 15 segundos, e deverão ser superiores a 0 °C de forma a validar as entradas, após isto os valores serão visualizados em tempo real no *Dashboard* do utilizador e colocados na base de dados dedicada, neste caso, será a *DB TMP Utente André Ferreira*.
- Caso a temperatura for superior a 37,6 °C ou inferior a 35 °C, é gerado um alerta e é enviado a respetiva notificação para o e-mail do familiar do utente.
- Todas as medições estarão disponíveis durante 24 horas, no final deste período, estas medições serão apagadas de forma permanente.

3.4.4 Flow para o registo dos alertas na base de dados

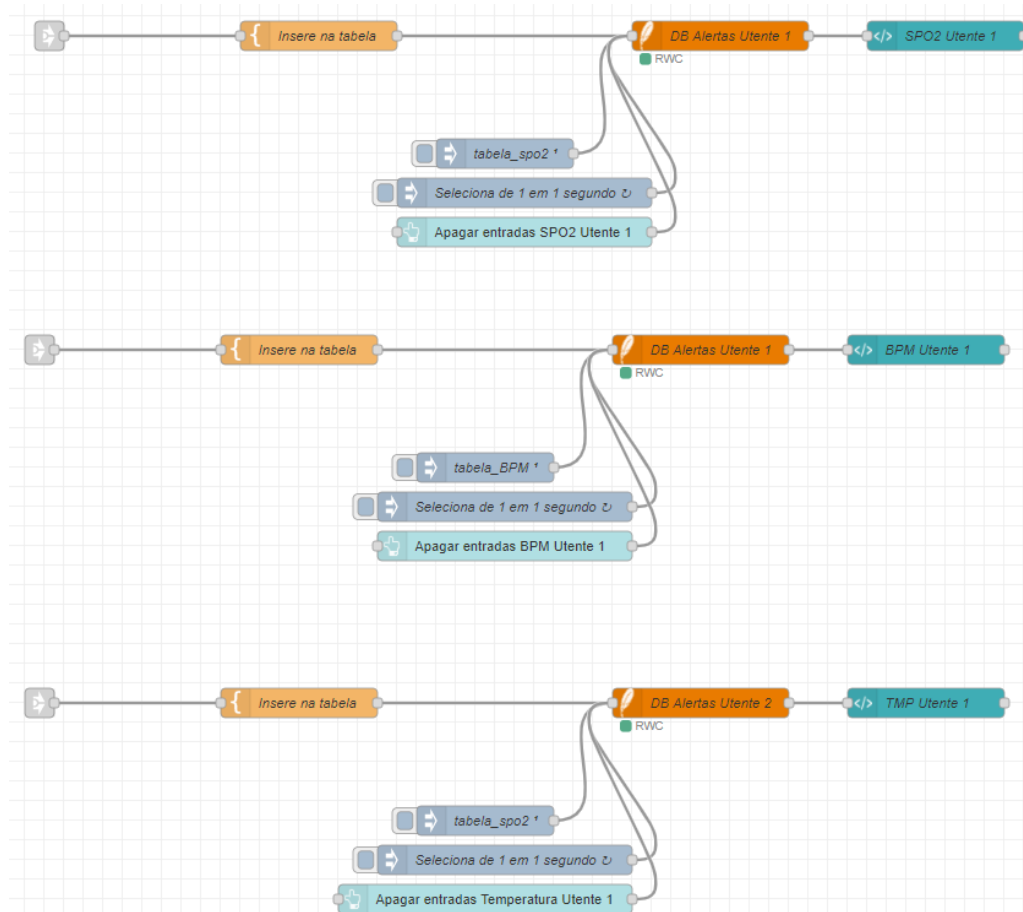


Figura 25 - Flow do processamento do registo de alertas de cada Utente

- Os Flows dos nós acima ilustrados, mostram o processo de gravação de leituras críticas dos respetivos valores de Saturação de oxigénio no sangue, frequência cardíaca e temperatura corporal.
- Ao contrário das leituras de valores considerados normais, estes valores críticos serão armazenados em bases de dados dedicadas (SPO2 Utente 1, BPM Utente1 e TMP Utente1) e serão mantidos de forma permanente, podendo, assim, ser possível consultar o histórico de cada utente ao longo do tempo.

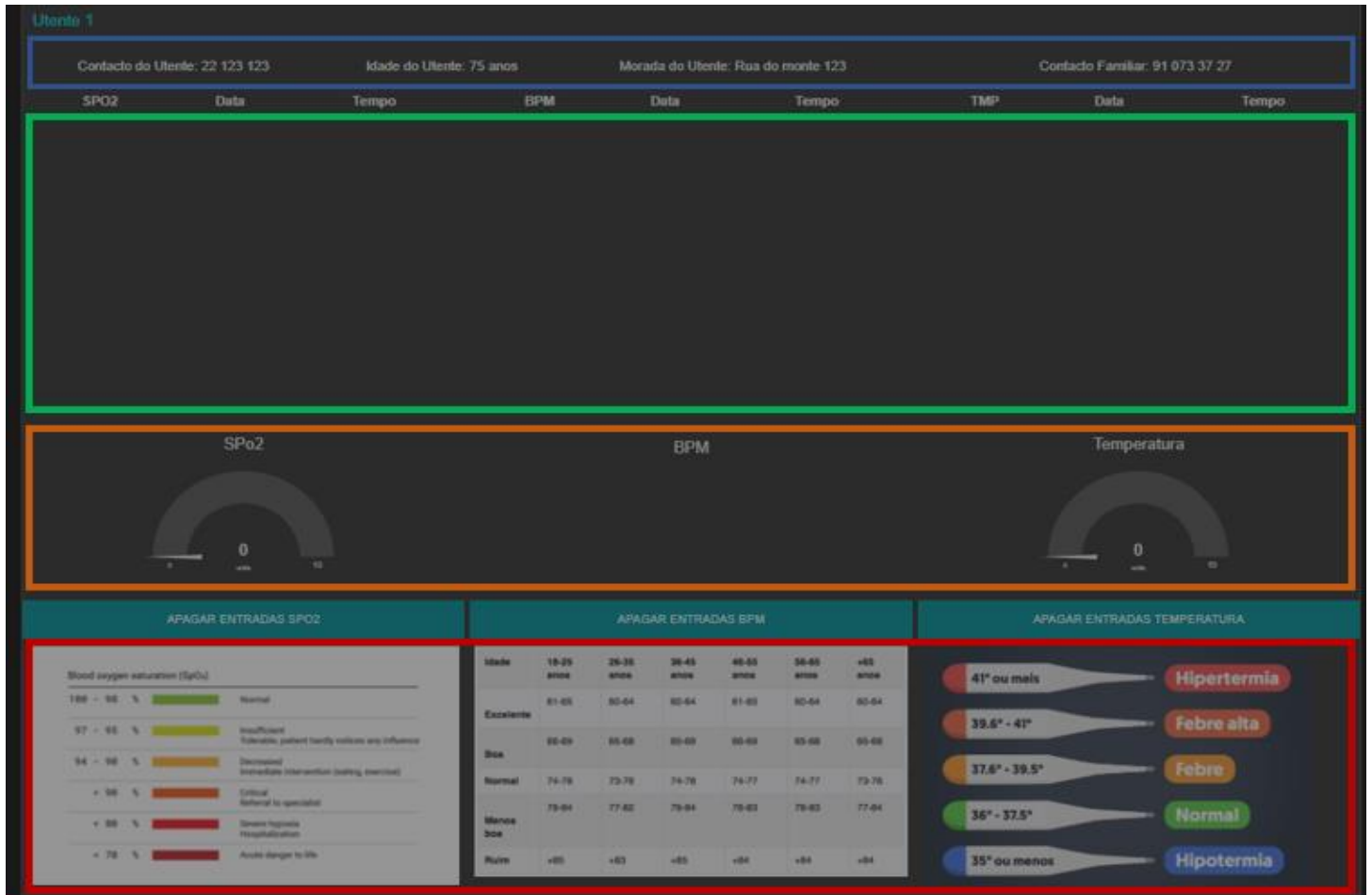
3.4.5 Nós para representação das informações na UI

A figura 26 abaixo, demonstra os nós responsáveis por guardar as informações dos utentes, que são visualizadas no *dashboard*, e também as imagens auxiliares que possuem os valores de referência para os parâmetros medidos.



Figura 26 - Nós relativos às informações dos utentes e às tabelas de valores ideais

3.4.6 Dashboard



- Visualização dos valores guardados nas bases de dados
- Manómetros e gráfico para medições em tempo real
- Imagens informativas sobre os valores ideais de cada parâmetro
- Dados do utente/idoso

Figura 27 – Vista geral do *dashboard* do utente e respetiva descrição das suas componentes

3.5 Resultados

Nesta demonstração foi utilizado o módulo do Utente 1 (com o sensor MAX30100). Depois de iniciar o módulo e do mesmo estabelecer ligação ao servidor na *Cloud*, coloca-se o dedo indicador sobre o sensor controlando a respiração para não influenciar os valores medidos, e aguarda-se cerca de 15 segundos para se obter uma leitura correta dos parâmetros. Na figura 28, pode observar-se a medição dos parâmetros da frequência cardíaca e de saturação de oxigénio, com recurso ao sensor MAX30100:

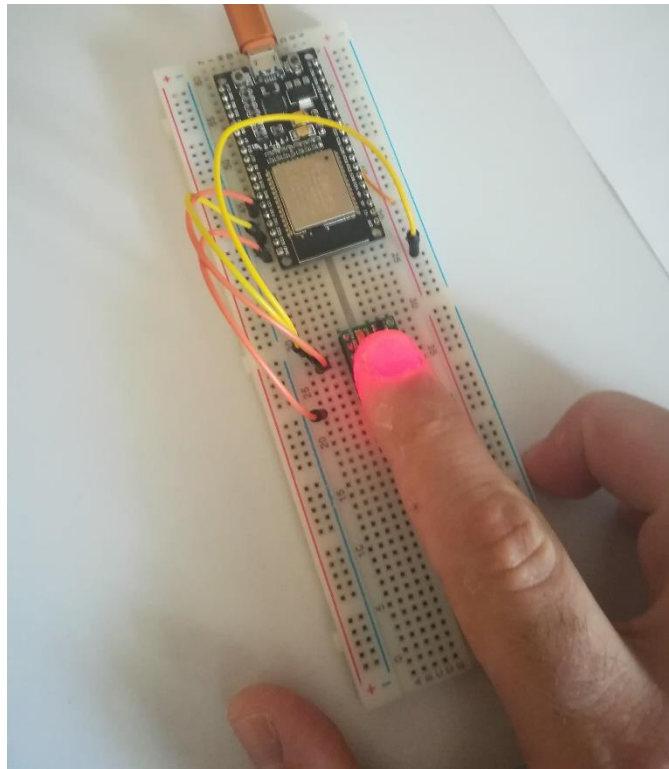


Figura 28 - Teste do módulo do Utente 1

Na figura 29 abaixo, temos um exemplo da visualização destes parâmetros medidos em tempo real no *dashboard* de interface do utilizador.



Figura 29 - Visualização dos valores na *interface* de utilizador

Do lado do servidor é possível visualizar em tempo real a medição que o Utente 1 está a fazer, no entanto os valores apenas são inseridos na base de dados com intervalos de 15 segundos, (tempo necessário para o sensor MAX30100 estabilizar as leituras) de forma que os valores guardados sejam o mais fiáveis possível. Além dos valores de SpO2 e BPM, também o registo da data e da hora das medições fica armazenada na base de dados. Após estes valores ficarem registados na base de dados, os mesmos permanecerão disponíveis para visualização por um período de 24 horas, no final deste período estes valores são apagados da base de dados, ficando as mesmas disponíveis para novas medições no dia seguinte.

Na figura 30 abaixo, simulamos uma leitura crítica dos valores de SpO2 no Utente 1. Ao acontecer isto (além do valor ficar registado na base de dados como visto anteriormente) aparece uma janela pop-up alertando que houve uma leitura crítica neste utente. Desta forma, a pessoa responsável pela monitorização destes idosos, pode imediatamente alertar o pronto-socorro para se dirigir à morada do utente (informação que se encontra no topo da *interface*).

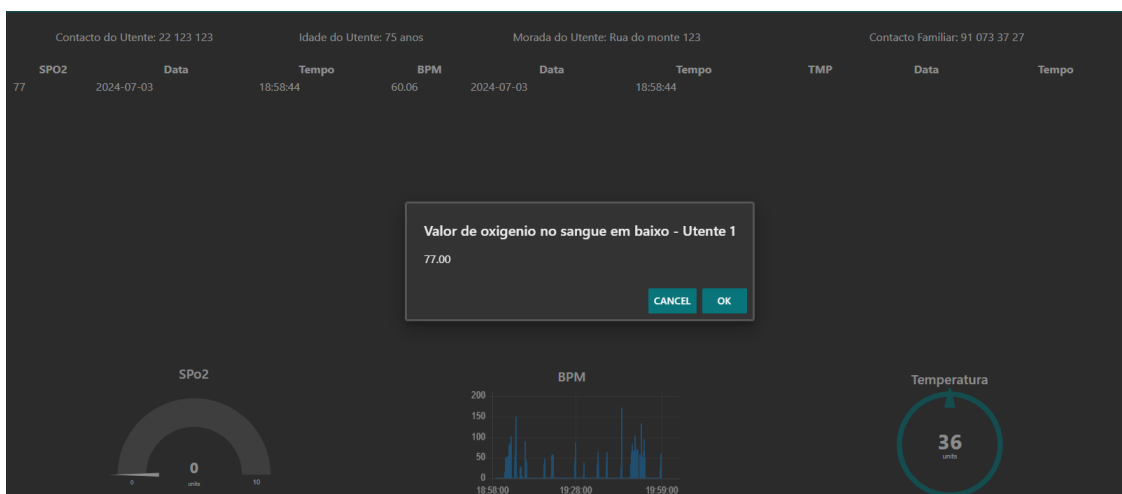


Figura 30 - Alerta gerado por leitura critica

Além do alerta gerado para o utilizador, também o familiar do idoso/utente recebe um e-mail com o assunto (URGENTE), a indicar que o seu familiar teve uma leitura crítica, e que foi reencaminhado para o hospital/posto de saúde mais próximo, como ilustrado na figura 31 abaixo.

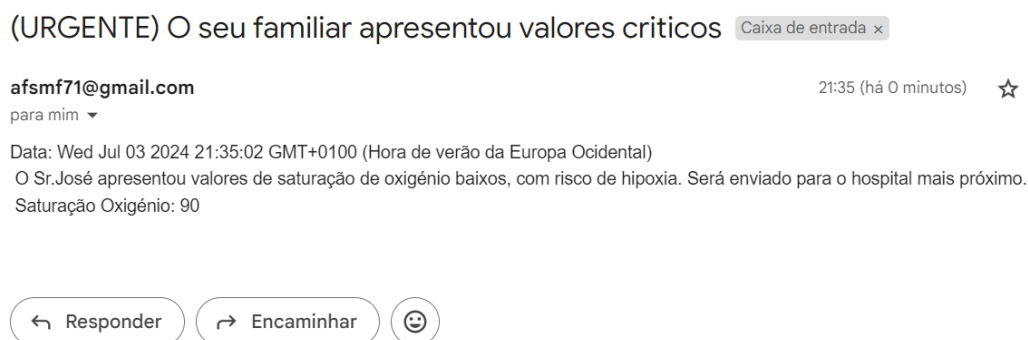


Figura 31 - Exemplo de e-mail enviado ao familiar do idoso/utente

Sempre que existir uma leitura crítica de qualquer parâmetro (neste caso trata-se do parâmetro SpO2) o valor fica registado numa base de dados que contém todos os alertas daquele utente. Ao contrário dos dados apresentados na *interface* do utente, estes dados apenas são eliminados caso o utilizador assim o deseje, caso contrário ficarão sempre guardados na base de dados. Isto permite manter um registo histórico ao longo do tempo de cada utente, de forma a isolar potenciais problemas de saúde crónicos (figura 32).

Alertas									
SPO2	Data	Tempo	BPM	Data	Tempo	TMP	Data	Tempo	
77	2024-07-03	18:58:44	105.11	2024-07-03	19:10:48				
90	2024-07-03	20:35:03	90.09	2024-07-03	20:12:09				
			86	2024-07-03	20:35:09				
APAGAR ENTRADAS SPO2 UTENTE 1			APAGAR ENTRADAS BPM UTENTE 1			APAGAR ENTRADAS TEMPERATURA UTENTE 1			
Utente 2									
SPO2	Data	Tempo	BPM	Data	Tempo	TMP	Data	Tempo	
90	2024-07-03	18:56:20	86	2024-07-03	18:56:22				

Figura 32 - Alertas guardados em base de dados própria

4 Modelo de negócio

Este projeto pode ser estruturado em torno de um modelo de negócio focado na prestação de um serviço de saúde de qualidade e que pode contribuir significativamente para melhorar a saúde e a qualidade de vida dos idosos em comunidades pequenas e isoladas, promovendo um sistema de saúde nacional mais eficaz e acessível.

1. Resumo executivo

O projeto propõe a criação de um sistema de monitorização de parâmetros de saúde de idosos em aldeias, utilizando módulos de leitura conectados a um servidor central. O objetivo é monitorizar o estado de saúde da população idosa, e proporcionar intervenções rápidas em casos de emergência.

2. Segmento de clientes

- Idosos que residem em aldeias.
- Familiares dos mesmos.
- Unidades de saúde e hospitais locais.
- Câmaras municipais ou entidades governamentais focadas em saúde pública.

3. Proposta de valor

- Aumentar a segurança e bem-estar dos idosos.
- Proporcionar monitorização contínua e em tempo real.
- Facilitar intervenções rápidas em situações críticas.
- Oferecer uma solução acessível e fácil de usar para a monitorização do estado de saúde geral.

4. Canais

- Parcerias com centros de saúde locais e hospitais.
- Divulgação através de redes sociais e eventos comunitários.
- Colaboração com organizações não-governamentais que atuam na área da saúde e bem-estar.

5. Relacionamento com clientes

- Suporte técnico para instalação e utilização dos módulos.
- Informação contínua sobre o estado de saúde dos idosos, através de relatórios para as respetivas famílias.
- Sessões de formação para os idosos sobre como usar os dispositivos e entender as leituras.

6. Fontes de receita

- Venda ou aluguer dos módulos de monitorização.
- Serviços de monitorização contínua (assinaturas mensais).
- Parcerias comerciais com entidades de saúde e seguradoras.
- Subvenções governamentais para projetos de saúde pública.

7. Recursos principais

- Tecnologia para a criação dos dispositivos (módulos de leitura).
- Plataforma online para adquirir e analisar os dados.
- Equipa médica para avaliação e resposta a situações críticas.
- Infraestrutura para logística de manutenção dos dispositivos.

8. Atividades principais

- Desenvolvimento e teste dos dispositivos de leitura.

- Monitorização e análise dos dados.
- Formação da equipa médica para resposta a emergências.
- Interação e apoio à comunidade local.

9. Parcerias principais

- Fabricantes de dispositivos médicos.
- Centros de saúde e hospitais da região.
- Universidades ou instituições de pesquisa para desenvolver tecnologia (ISEP).
- Organizações locais e entidades governamentais para apoio ao projeto.

10. Estrutura de custos

- Custos de produção e manutenção dos dispositivos.
- Salários para a equipe técnica e médica.
- Despesas de marketing e comunicação com a população.
- Custos operacionais do servidor.

4.1 Custos

Nas tabelas seguintes, podemos verificar o custo estimado para a criação de um módulo de leitura e o custo de implementação do servidor.

Tabela 3 - Custos do módulo do utente

Módulo do utente	
ESP32	11,00 €
Sensor MAX30100	7,32 €
Sensor Temperatura	6,00 €
Caixa selada 3D	3,00 €
Total (1 unidade)	27,32 €
Total (1000 unidades)	2 732,00 €

Tabela 4 - Custos do servidor

Servidor	
Raspberry Pi	63,00 €
Computador desktop completo	400,00 €
Brooker online plano anual	300,00 €
Total	763,00 €

Estratégias para Reduzir Custos

1. Comprar em Lote

- **Economia de escala:** Adquirir os componentes dos módulos de leitura em quantidade pode reduzir o custo por unidade. Os fabricantes costumam oferecer descontos significativos para compras em grande escala.

2. Parcerias com Universidades ou centros de investigação

- **Desenvolvimento colaborativo:** Colaborar com universidades ou centros de investigação pode trazer inovação no *design* do produto, potencialmente reduzindo custos. Os estudantes podem realizar estágios ou projetos finais que incluem a construção dos módulos.

3. Explorar fornecedores alternativos

- **Cotação de preços:** Pesquisar diferentes fornecedores e negociar preços melhores. Às vezes, a troca de fornecedores pode resultar em economias significativas.

4. Uso de tecnologia de código aberto

- **Software Livre:** Utilizar plataformas e softwares de código aberto para o servidor pode reduzir custos com licenças de software. Isso pode incluir o uso de sistemas operativos, bases de dados e frameworks gratuitos.

5. Otimização dos recursos do servidor

- **Infraestrutura em nuvem:** Considerar a utilização de serviços em nuvem em vez de servidores físicos, o que em muitos casos pode resultar na mitigação dos custos em termos de hardware e manutenção.

6. Módulos de leitura multifuncionais

- **Redução das funcionalidades:** Analisando as funcionalidades necessárias, pode-se optar por desenvolver um único módulo que integre diversas funcionalidades (frequência cardíaca, saturação de oxigênio, temperatura corporal) em vez de componentes separados, reduzindo assim os custos de componentes e fabrico.

7. Procura de subsídios e financiamentos

- **Apoios governamentais:** Procurar por subsídios e financiamentos disponíveis para projetos de saúde pública. Com alguma frequência, existem fundos que estão disponíveis para iniciativas que procurem melhorar a saúde da população.

8. Minimização da manutenção

- **Design para durabilidade:** Investir tempo na pesquisa e desenvolvimento para criar módulos mais fiáveis e fáceis de manter. Isto pode resultar na redução de custos de manutenção a longo prazo.

9. Feedback da comunidade

- **Iteração de produto:** Inquirir a comunidade na fase de testes do produto, pode proporcionar comentários que ajudem a simplificar o projeto do módulo e cortar custos desnecessários por funcionalidades não essenciais.

10. Utilização de tecnologias existentes

- **Reuso de tecnologias:** Avaliar o uso de tecnologias já existentes (sensores, plataformas de monitorização) que podem ser adaptadas ou reaproveitadas para este projeto, reduzindo assim o custo de desenvolvimento.

Ao implementar estas estratégias, poderá ser possível reduzir os custos iniciais de produção e manutenção, mas também desenvolver um projeto mais eficiente e sustentável a longo prazo.

4.2 Projeto Piloto

Ideia para projeto piloto em Portugal

Localização: Selecionar uma aldeia em Portugal com uma percentagem significativa de população idosa.

Duração do projeto: 6 meses.

Fases do projeto:

1. **Identificação de parceria:** Colaboração com um centro de saúde local para identificar os idosos que beneficiarão do programa.
2. **Instalação dos dispositivos:** Fornecimento dos módulos de monitorização para cada idoso identificado. Realizar sessões de formação para ensinar como usar os dispositivos.
3. **Monitorização diária dos dados:** Coleta de dados em tempo real ou por agendamento telefónico, e análise das leituras dos parâmetros de saúde.
4. **Respostas a crises:** Implementação de um protocolo de resposta a emergências, em que, em caso de leituras críticas, uma equipa de saúde pode ser acionada rapidamente.
5. **Avaliação e relatório final:** Elaboração de um relatório com os resultados e comentários da comunidade sobre a eficácia do projeto e propostas de melhorias.

5 Conclusões

Em jeito de conclusão, os resultados obtidos neste trabalho demonstraram que os principais objetivos propostos foram alcançados com sucesso, oferecendo uma solução eficiente para a monitorização do estado de saúde geral em comunidades isoladas. A criação de um sistema robusto, possibilitou o envio seguro e confiável dos dados para um servidor central através do protocolo MQTT, este protocolo garantiu a integridade da comunicação, enquanto a tecnologia GPRS, utilizando a rede 2G, assegurou a conectividade em áreas remotas, sem infraestrutura Wi-Fi.

Além disso, a implementação de um *dashboard* para visualização e tratamento dos dados possibilitou a monitorização em tempo real por profissionais de saúde, permitindo uma intervenção rápida e eficaz. A introdução de alertas críticos, que notificam tanto a família do idoso quanto a equipa de emergência local, assegura uma resposta proativa em situações de risco, melhorando a segurança e o bem-estar dos indivíduos monitorizados.

5.1 Trabalhos futuros

Em termos de trabalhos futuros, existem várias possibilidades de aprimoramento. A adição de mais sensores para capturar um leque mais amplo de parâmetros de saúde permitiria uma monitorização ainda mais completa e precisa. O aprimoramento do *dashboard*, tornando-o mais intuitivo e *user-friendly* para o profissional de saúde, também contribuiria para a eficiência no acompanhamento dos dados. Por fim, a melhoria no módulo de sensores, com o intuito de facilitar a sua utilização por parte dos idosos, seria um passo importante para garantir uma maior adesão ao sistema e melhorar a qualidade da resposta medica em zonas remotas de Portugal.

Durante a execução deste projeto, acredito que me comprometi a 100 % com os objetivos e as metas que estipulei desde o início. Existiram, sem dúvida, contratempos e obstáculos de ordem profissional, mas que no final acabaram por ser mitigados, fruto também da dedicação e ajuda do meu orientador, o professor Dr. Joaquim Alves, e da troca de ideias entre colegas de curso que é essencial para a aprendizagem. Com este projeto procurei incluir conceitos lecionados ao longo deste Mestrado, não sendo possível, no entanto, abordar todos os temas, mas pelo menos aqueles que maior interesse despertou em mim. A preocupação com o aumento do isolamento em Portugal, fez-me tomar a decisão de fazer algo que pudesse ser uma possível solução para este cenário. Esta área tem despertado também alguma curiosidade juntamente com o interesse em aprender mais sobre as tecnologias usadas na medicina, e compreender de que forma é possível melhorar ou inovar neste campo. Por fim, acredito que este projeto tem grande potencial de implementação no futuro, tendo em conta as vantagens que pode trazer no combate aos efeitos adversos provocados pelo isolamento, e a melhoria da segurança e da qualidade de vida destes idosos, e também, da eficácia do Sistema Nacional de Saúde. Ao longo do meu percurso no Instituto Superior de Engenharia do Porto, fui-me enriquecendo não só através do que aprendi, mas também das pessoas que conheci, desde alunos a professores, todos tiveram de certa forma envolvidos na minha evolução como profissional e como pessoa. A todos estes, as minhas saudações académicas e votos de um futuro promissor.

Referências

- [1] «Censos 2011-Resultados Definitivos».
- [2] I. Com, «Global strategy and action plan on ageing and health», 2017, Acedido: 26 de Novembro de 2024. [Em linha]. Disponível em: <http://apps.who.int/bookorders>.
- [3] D. Jardim, J. Gil, F. Martins, L. Ferreira, e R. Fonseca-Pinto, «Sistema de aquisição e registo de sinais biomédicos baseado numa plataforma de desenvolvimento open-source de baixo custo». [Em linha]. Disponível em: <http://cdrsp.ipleiria.pt>
- [4] Cristiana Pinto, Pedro Moreira, Rita Vilaça, e Rui Guedes, «Monitorização de Idosos Isolados».
- [5] S. Banka, I. Madan, e S. S. Saranya, «Smart Healthcare Monitoring using IoT», 2018. [Em linha]. Disponível em: <http://www.ripublication.com>
- [6] «Teach, learn, and make with the Raspberry Pi Foundation». Acedido: 29 de Setembro de 2024. [Em linha]. Disponível em: <https://www.raspberrypi.org/>
- [7] Y. G. Lim *et al.*, «Monitoring physiological signals using nonintrusive sensors installed in daily life equipment», *Biomed Eng Lett*, vol. 1, n. 1, pp. 11–20, Fev. 2011, doi: 10.1007/S13534-011-0012-0/METRICS.
- [8] «O que é Taquicardia | Medtronic». Acedido: 26 de Novembro de 2024. [Em linha]. Disponível em: <https://www.medtronic.com/br-pt/your-health/conditions/fast-heart-beat.html>
- [9] «Hipóxia: o que é, sintomas, causas e tratamento - Tua Saúde». Acedido: 26 de Novembro de 2024. [Em linha]. Disponível em: <https://www.tuasaude.com/hipoxia/>
- [10] A. Ferreira Da Rocha *et al.*, «Processamento de Sinais Biológicos».
- [11] «Frequência cardíaca normal: batimento cardíaco por idade - Tua Saúde». Acedido: 29 de Setembro de 2024. [Em linha]. Disponível em: <https://www.tuasaude.com/frequencia-cardiaca/>
- [12] «Oxímetro: para que serve e como usar corretamente - Tua Saúde». Acedido: 29 de Setembro de 2024. [Em linha]. Disponível em: <https://www.tuasaude.com/oximetria/>
- [13] «What is Oxygen Saturation?» Acedido: 26 de Novembro de 2024. [Em linha]. Disponível em: <https://www.news-medical.net/health/What-is-Oxygen-Saturation.aspx>
- [14] M. W. Sjoding, R. P. Dickson, T. J. Iwashyna, S. E. Gay, e T. S. Valley, «Racial Bias in Pulse Oximetry Measurement», *New England Journal of Medicine*, vol. 383, n. 25, pp. 2477–2478, Dez. 2020, doi: 10.1056/NEJMC2029240.

- [15] K. H. Shelley, «Photoplethysmography: beyond the calculation of arterial oxygen saturation and heart rate», *Anesth Analg*, vol. 105, n. 6 Suppl, 2007, doi: 10.1213/01.ANE.0000269512.82836.C9.
- [16] B. P. Santos *et al.*, «Internet das Coisas: da Teoria à Prática».
- [17] V. Bhatt e S. Chakraborty, «Improving service engagement in healthcare through internet of things based healthcare systems», *Journal of Science and Technology Policy Management*, vol. 14, n. 1, pp. 53–73, Fev. 2023, doi: 10.1108/JSTPM-03-2021-0040/FULL/XML.
- [18] «ESP32 With Onboard SIM800L ! » DIY Usthad». Acedido: 29 de Setembro de 2024. [Em linha]. Disponível em: <https://diyusthad.com/2019/07/esp32-with-onboard-sim800.html>
- [19] «ESP32 Series Datasheet Version 4.7 2.4 GHz Wi-Fi + Bluetooth[®] + Bluetooth LE SoC Including». [Em linha]. Disponível em: www.espressif.com
- [20] «7 Maiores Desafios em Projetos de IoT em 2023 e Como Resolvê-los | Blog da emnify». Acedido: 25 de Novembro de 2024. [Em linha]. Disponível em: <https://www.emnify.com/pt-br/blog/desafios-iot-2023>
- [21] «TTGO T-Call V1.3 ESP32 Pinout » DIY Usthad». Acedido: 25 de Novembro de 2024. [Em linha]. Disponível em: <https://diyusthad.com/2020/08/ttgo-t-call-v1-3-esp32-pinout.html>
- [22] «In-Depth: Send Receive SMS & Call with SIM800L GSM Module & Arduino». Acedido: 29 de Setembro de 2024. [Em linha]. Disponível em: <https://lastminuteengineers.com/sim800l-gsm-module-arduino-tutorial/>
- [23] «How To Play with SIM800L: Part 1». Acedido: 29 de Setembro de 2024. [Em linha]. Disponível em: <https://www.electroschematics.com/introducingsim800l/>
- [24] «Datasheet_SIM800L».
- [25] «Sim800L GSM module pinout (as adopted from... | Download Scientific Diagram». Acedido: 25 de Novembro de 2024. [Em linha]. Disponível em: https://www.researchgate.net/figure/Sim800L-GSM-module-pinout-as-adopted-from_fig6_361500700
- [26] «General Description». [Em linha]. Disponível em: www.maximintegrated.com
- [27] «RCWL-0530/MAX30100 an oximetry / heart rate / temperature sensor – Peko’s Library». Acedido: 25 de Novembro de 2024. [Em linha]. Disponível em: <https://pekolibrary.wordpress.com/2018/05/27/rcwl-0530-max30100/>
- [28] «MLX90614 family Datasheet Single and Dual Zone Infra Red Thermometer in TO-39 Features and Benefits».

- [29] «MLX90614 Contactless IR Temperature Sensor Interfacing with Arduino». Acedido: 25 de Novembro de 2024. [Em linha]. Disponível em: <https://microcontrollerslab.com/mlx90614-contactless-ir-temperature-sensor-pinout-interfacing-arduino/>
- [30] «O que é um Servidor Cloud? Como funcionam os Servidores Cloud? | Microsoft Azure». Acedido: 25 de Novembro de 2024. [Em linha]. Disponível em: <https://azure.microsoft.com/pt-pt/resources/cloud-computing-dictionary/what-is-a-cloud-server>
- [31] «Arquitetura de computação em nuvem Diagrama de rede de computadores, aplicativo Web, rede de computadores, texto, serviço png | PNGWing». Acedido: 25 de Novembro de 2024. [Em linha]. Disponível em: <https://www.pngwing.com/pt/free-png-npvce>
- [32] «Servidor em nuvem | Dell BRASIL». Acedido: 29 de Setembro de 2024. [Em linha]. Disponível em: <https://www.dell.com/pt-br/dt/learn/cloud/cloud-server.htm>
- [33] «MQTT Version 3.1.1». Acedido: 26 de Novembro de 2024. [Em linha]. Disponível em: <https://docs.oasis-open.org/mqtt/mqtt/v3.1.1/os/mqtt-v3.1.1-os.html>
- [34] «MQTT – Wikipédia, a enciclopédia livre». Acedido: 25 de Novembro de 2024. [Em linha]. Disponível em: <https://pt.wikipedia.org/wiki/MQTT>
- [35] «Comunicação I2C - Embarcados - Sua fonte de informações sobre Sistemas Embarcados». Acedido: 25 de Novembro de 2024. [Em linha]. Disponível em: <https://embarcados.com.br/comunicacao-i2c/>
- [36] «O que é mqtt broker e quais as suas principais vantagens?» Acedido: 25 de Novembro de 2024. [Em linha]. Disponível em: <https://engprocess.com.br/mqtt-broker/>
- [37] «Eclipse Mosquitto». Acedido: 25 de Novembro de 2024. [Em linha]. Disponível em: <https://mosquitto.org/>
- [38] «What is Mosquitto MQTT?» Acedido: 25 de Novembro de 2024. [Em linha]. Disponível em: <https://www.eginnovations.com/documentation/Mosquitto-MQTT/What-is-Mosquitto-MQTT.htm>
- [39] «Introdução ao Node-RED. Introdução | by Vicjrgon | Medium». Acedido: 26 de Novembro de 2024. [Em linha]. Disponível em: <https://medium.com/@vicjrgon/introdu%C3%A7%C3%A3o-ao-node-red-230db5c8d163>

Anexos

Programa módulo do utente 1 (Wi-Fi)

```
2  #include <Wire.h>
3  #include "MAX30100_PulseOximeter.h"
4  #include <PubSubClient.h>
5  #include <WiFi.h>
6
7  #define REPORTING_PERIOD_MS 1000
8
9
10 // _____
11
12 // Lista de redes Wi-Fi
13
14 const char* ssid = "Vodafone-AE62F7";
15 const char* password = "AGA3HCAEGXXXGCJC";
16
17 //Rede casa
18 //SSID: Vodafone-AE62F7
19 //password: AGA3HCAEGXXXGCJC
20
21 // _____
22 MAX30100 sensor;
23 PulseOximeter pox;
24 uint32_t tsLastReport = 0;
25 unsigned long previousMillis = 0;
26 unsigned long interval = 30000;
```

```

--
29 // Endereço IP do RPI
30 const char* mqtt_server = "192.168.1.150";
31
32 //192.168.1.150 casa
33
34 // _____
35
36 // Inicializa o espClient
37
38 WiFiClient espClient;
39 PubSubClient client(espClient);
40
41 // _____
42
43 // Esta função presente na biblioteca Wifi.h permite ao esp32 ligar-se a uma rede Wi-fi
44
45 void setup_wifi() {
46     delay(10);
47     // We start by connecting to a WiFi network
48     Serial.println();
49     Serial.print("Connecting to ");
50     Serial.println(ssid);
51     WiFi.begin(ssid, password);
52     while (WiFi.status() != WL_CONNECTED) {
53         delay(500);
54         Serial.print(".");
55     }
56     Serial.println("");
57     Serial.print("WiFi connected - ESP IP address: ");
58     Serial.println(WiFi.localIP());
59 }
60

```

```

// Esta função reconecta o ESP32 ao MQTT broker
// Permite também subscrever a mais topicos com o ESP32

void reconnect() {

  // Fica em Loop até nova reconexão
  while (!client.connected()) {
    Serial.print("Attempting MQTT connection...");

    if (client.connect("Utente_1", mqttUsername, mqttPassword)) {
      Serial.println("connected");

      // Subscrever ou Resubscrever um topico

      //client.subscribe("ESP32/SPo2");
      //client.subscribe("ESP32/BPM");
    } else {
      Serial.print("failed, rc=");
      Serial.print(client.state());
      Serial.println(" try again in 5 seconds");
      // Espera 5 segundos antes de tentar novamente
      delay(5000);
    }
  }
}

void setup()
{
  Serial.begin(9600);

  setup_wifi();

  client.setServer(mqtt_server, 1883);

  Serial.print("Enabling HR/SPO2 mode..");
  sensor.setMode(MAX30100_MODE_SPO2_HR);
  Serial.println("done.");

  Serial.print("Initializing pulse oximeter..");

  if (!pox.begin()) {
    Serial.println("FAILED");
    for(;;);
  }
  else {
    Serial.println("SUCCESS");
  }

  pox.setIRLedCurrent(MAX30100_LED_CURR_14_2MA);
  //sensor.setLedsCurrent(MAX30100_LED_CURR_14_2MA, MAX30100_LED_CURR_14_2MA);
  //MAX30100_LED_CURR_14_2MA
  //MAX30100_LED_CURR_7_6MA
}

```

```

void loop()
{
    if (!client.connected()) {
        reconnect();
    }

    if(!client.loop()) {
        client.connect("Utente_1");
    }

    unsigned long currentMillis = millis();
    // if WiFi is down, try reconnecting every CHECK_WIFI_TIME seconds
    if ((WiFi.status() != WL_CONNECTED) && (currentMillis - previousMillis >= interval)) {
        Serial.print(millis());
        Serial.println("Reconnecting to WiFi...");
        WiFi.disconnect();
        WiFi.reconnect();
        previousMillis = currentMillis;
    }

    // Esta função deve ser chamada o mais rapido possível
    pox.update();

    //De forma assincrona coloca os valores de SPO2 e BPM na porta serie
    //Para estes 2 parametros valores de 0 significam "Invalido"

    if (millis() - tsLastReport > REPORTING_PERIOD_MS)
    {
        static char valor_SPO2[7];
        dtostrf(pox.getSpO2(),6,2, valor_SPO2);

        static char valor_BPM[7];
        dtostrf(pox.getHeartRate(),6,2, valor_BPM);

        // Publica os valores de Spo2 e Epm

        client.publish("Utente_1/SPO2",valor_SPO2);
        client.publish("Utente_1/BPM", valor_BPM);

        Serial.print("Heart rate:");
        Serial.print(pox.getHeartRate());
        Serial.print("bpm / SpO2:");
        Serial.print(pox.getSpO2());
        Serial.println("%");
        tsLastReport = millis();
    }
}
}

```

Programa módulo do utente 2 (GPRS)

```
// Seleção do modem:
#define TINY_GSM_MODEM_SIM800 // SIM800L

// Definir porta serie para consola de depuração (para o Monitor Serial, velocidade padrão 115200)
#define SerialMon Serial
// Definir porta serie para comandos AT
#define SerialAT Serial1

// Definir a porta serie para impressões de depuração, se necessário
#define TINY_GSM_DEBUG SerialMon

// Pin GSM
#define GSM_PIN ""

//Credenciais GSM
const char apn[] = "net2.vodafone.pt"; // APN
const char gprsUser[] = "vodafone";
const char gprsPass[] = "vodafone";

// SIM card PIN
const char simPIN[] = "7145";

// MQTT detalhes
const char* broker = "165.227.234.92"; // Endereço de IP publico
#define MQTT_PORT 1883

// Definir a porta serie para impressões de depuração, se necessário
// #define DUMP_AT_COMMANDS

#include <Wire.h>
#include <TinyGsmClient.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_MLX90614.h>

#ifdef DUMP_AT_COMMANDS
    #include <StreamDebugger.h>
    StreamDebugger debugger(SerialAT, SerialMon);
    TinyGsm modem(debugger);
#else
    TinyGsm modem(SerialAT);
#endif

#include <PubSubClient.h>

TinyGsmClient client(modem);
PubSubClient mqtt(client);

// TTGO T-Call pins
#define MODEM_RST          5
#define MODEM_PWKEY        4
#define MODEM_POWER_ON    23
#define MODEM_TX            27
#define MODEM_RX            26

Adafruit_MLX90614 mlx = Adafruit_MLX90614();
```

```

uint32_t lastReconnectAttempt = 0;

long lastMsg = 0;

boolean mqttConnect() {
    SerialMon.print("Connecting to ");
    SerialMon.print(broker);

    //Conectar ao MQTT Broker sem nome de usuário e senha
    //boolean status = mqtt.connect("Utente_1");

    // Ou, com autenticação:
    boolean status = mqtt.connect("GsmClientN", mqttUsername, mqttPassword);

    if (status == false) {
        SerialMon.println(" fail");
        ESP.restart();
        return false;
    }
    SerialMon.println(" success");
    //mqtt.subscribe(topicOutput1);
    //mqtt.subscribe(topicOutput2);

    return mqtt.connected();
}

void setup() {
    // Define o baud rate da porta serie
    SerialMon.begin(115200);
}

```

```

delay(10);
mlx.begin();
Wire.begin();

// Definir redefinição do modem, habilitar pinos de alimentação
pinMode(MODEM_PWKEY, OUTPUT);
pinMode(MODEM_RST, OUTPUT);
pinMode(MODEM_POWER_ON, OUTPUT);
digitalWrite(MODEM_PWKEY, LOW);
digitalWrite(MODEM_RST, HIGH);
digitalWrite(MODEM_POWER_ON, HIGH);

// Definir a taxa de transmissão do módulo GSM e pinos UART
SerialAT.begin(115200, SERIAL_8N1, MODEM_RX, MODEM_TX);
delay(6000);

SerialMon.println("Initializing modem...");
modem.restart();
// modem.init();

String modemInfo = modem.getModemInfo();
SerialMon.print("Modem Info: ");
SerialMon.println(modemInfo);

// Desbloqueia o cartão SIM com um PIN, se necessário
if ( GSM_PIN && modem.getSimStatus() != 3 ) {
    modem.simUnlock(GSM_PIN);
}

```

```

SerialMon.print("Connecting to APN: ");
SerialMon.print(apn);
if (!modem.gprsConnect(apn, gprsUser, gprsPass)) {
    SerialMon.println(" fail");
    ESP.restart();
}
else {
    SerialMon.println(" OK");
}

if (modem.isGprsConnected()) {
    SerialMon.println("GPRS connected");
}

// MQTT Broker setup
mqtt.setServer(broker, 1883);

void loop() {

    if (!mqtt.connected()) {
        SerialMon.println("=== MQTT NOT CONNECTED ===");
        // Reconnecta a cada 10 segundos
        uint32_t t = millis();
        if (t - lastReconnectAttempt > 10000L) {
            lastReconnectAttempt = t;
            if (mqttConnect()) {
                lastReconnectAttempt = 0;
            }
        }
        delay(100);
        return;
    }

    static char valor_TMP[7];
    dtostrf(mlx.readObjectTempC(), 6, 2, valor_TMP);

    // Publica os valores da temperatura
    mqtt.publish("Utente_2/TEMP", valor_TMP);
    Serial.print("*C\tObject = "); Serial.print(mlx.readObjectTempC()); Serial.println("*C");

    mqtt.loop();
}

```