

Sistema para Monitoramento Remoto de Sinais Vitais Utilizando ESP32

Anderson Ribeiro Ribas¹, Alessandro André Mainardi de Oliveira¹

¹Curso de Bacharelado em Ciência da Computação – Universidade Franciscana – 97010-032 – Santa Maria – RS – Brasil

a.ribas@ufn.edu.br, alessandroandre@ufn.edu.br

Abstract. *The importance of health in contrast to the critical scenario generated by COVID-19 requires the design of appropriate solutions for the preservation of life. From this perspective, the present work presents the development of a system for the remote monitoring of vital signs of patients in home health care. Heart rate, blood oxygen saturation and body temperature were observed. To acquire these parameters, the Max30100 and LM35 sensors are used. Whose data has been sent to the ESP32 microcontroller. Under the MQTT protocol, this information will be available on the web. Thus, even at a distance, the medical team will be able to monitor the patient's vital signs.*

Resumo. *A importância da saúde em contraste com o cenário crítico gerado pela COVID-19 exige a concepção de soluções adequadas à preservação da vida. Sob esta ótica, o presente trabalho apresenta o desenvolvimento de um sistema para o monitoramento remoto de sinais vitais de pacientes em atendimento domiciliar à saúde. Foram observadas a frequência cardíaca, a saturação de oxigênio no sangue e a temperatura corporal. Para aquisição desses parâmetros, são utilizados os sensores Max30100 e LM35. Cujos dados foram enviados ao microcontrolador ESP32. Sob o protocolo MQTT estas informações ficarão disponíveis na web. Assim, mesmo à distância, a equipe médica poderá acompanhar os sinais vitais do doente.*

Palavras-chaves: *Atendimento à saúde, ESP32, IoT, Monitoramento Remoto, MQTT, Sinais Vitais*

1. Introdução

A vida do ser humano é composto por um conjunto de condições que a define e possibilita a interação do corpo físico com o espaço-tempo em um período finito. As circunstâncias para a existência de vida são chamadas de sinais vitais, os quais podem ser medidos e seus parâmetros refletem as condições de saúde de uma pessoa.

Entre os sinais vitais, destacam-se a temperatura corporal, o pulso, a pressão arterial (PA), a frequência respiratória e a saturação de oxigênio no sangue (SpO₂), pois são comumente avaliados pela equipe médica. Ao monitorar esses sinais, é possível identificar a condição de saúde do indivíduo, ou seja, observar a eficiência dos sistemas imunológicos, cardiovascular, endócrino, respiratório e nervoso.

Uma mudança nos sinais vitais indica uma mudança na função fisiológica. A determinação dos sinais vitais fornece dados para identificar diagnósticos de enfermagem, para implantar intervenções planejadas e para avaliar os resultados da assistência. Uma alteração dos sinais vitais sinaliza a necessidade de intervenção médica ou da enfermagem.

Os sinais vitais são um modo eficiente e rápido de monitorar a condição do cliente ou de identificar problemas e avaliar a resposta do cliente a uma intervenção [Potter e Perry 2009, p.503].

Verifica-se a relevância destes sinais para a manutenção da saúde, e a verificação destes sinais podem acontecer em atendimentos clínicos hospitalares ou em atendimentos domiciliares. Este último é chamado de Atenção Domiciliar à Saúde e suas características serão consideradas para a realização deste trabalho.

De acordo com Duarte e Diogo (2000), os principais motivos que levaram ao surgimento do formato domiciliar no Brasil, foram: O envelhecimento populacional cada vez mais acentuado; o aumento de doenças crônicas não transmissíveis; os custos do sistema hospitalar cada vez mais elevados; o desenvolvimento de equipamentos tecnológicos; o aumento da procura por cuidados de saúde; o interesse dos profissionais da saúde por novas áreas de atuação; a exigência por maior privacidade, individualização e humanização da assistência à saúde, além da necessidade de maior integração da equipe profissional com o cliente e sua família.

Logo, o êxito do Atendimento Domiciliar à Saúde depende da adaptação desta modalidade de cuidado ao cenário socioeconômico apresentado pelo paciente. Por conseguinte, é indispensável disponibilizar aparelhos e equipamentos que permitam o diagnóstico médico em diversos ambientes. Além disso, Barro et al. (2002), ressaltam a importante relação entre a evolução dos equipamentos de monitoramento e o aumento na qualidade dos cuidados na saúde. Neste sentido, utilizar a tecnologia para fabricar um equipamento capaz de monitorar remotamente os sinais vitais do cliente, garantiria o sucesso neste tipo de atendimento.

Em 30 de janeiro de 2020, a Organização Mundial de Saúde (OMS), declarou que o surto da doença causada pelo novo coronavírus a COVID-19, constitui uma Emergência de Saúde Pública de importância Internacional - O mais alto nível de alerta da organização. No dia 11 de março de 2020, o surto da COVID-19 foi caracterizado pela OMS como uma pandemia.

Segundo a OMS (2020), até o dia 20 de maio de 2020 foram confirmados 4 801 202 casos e 318 935 mortes. Sendo que até esse momento não há vacinas contra o vírus, e a única forma de conter a proliferação da doença é o isolamento, o distanciamento social e cuidados com a higiene.

Diante desse cenário, viabilizar um dispositivo para monitorar sinais vitais no atendimento domiciliar à saúde, é uma alternativa para mitigar a ação do vírus impedindo o contágio da doença, bem como, evitando internações hospitalares desnecessárias. Com base na *Internet das Coisas* (do inglês, *Internet of Things (IoT)*), na utilização do microcontrolador ESP32 e nos sensores de temperatura, frequência cardíaca e saturação de oxigênio no sangue será possível criar um equipamento que auxilie a equipe médica nos cuidados com o paciente.

1.1. Objetivo Geral

O objetivo deste trabalho é construir um protótipo para monitorar os sinais vitais do paciente no atendimento domiciliar à saúde. Serão adquiridos os valores da temperatura corporal, do batimento cardíaco e do nível de oxigênio no sangue, cujos dados serão enviados imediatamente à equipe de saúde ou médico responsável.

1.2. Objetivos Específicos

- ✓ Construir um protótipo da estrutura do dispositivo para monitoramento dos sinais vitais;
- ✓ Realizar a aquisição de dados dos sensores do tipo LM35 e Max30100 conectados ao microcontrolador ESP32;
- ✓ Implementar a comunicação do ESP32 com a plataforma *ThingSpeak*.

2. Referencial Bibliográfico

Com o intuito de apresentar esta seção de forma clara e organizada, foi necessário dividi-la nos seguintes tópicos: conceitos, *hardwares* e protocolo.

2.1. Conceitos

Nesta subseção, serão apresentados os conceitos necessários para o desenvolvimento deste projeto.

2.1.1. Atenção domiciliar à saúde

A atenção domiciliar à saúde é definida como um termo genérico, que envolve ações de promoção à saúde, prevenção e tratamento de doenças e reabilitação, desenvolvidas em domicílio [Brasil, 2006]. A atenção domiciliar abrange três modalidades: O atendimento domiciliar, a internação domiciliar e a visita domiciliar.

A assistência domiciliar abrange tanto atividades simples como as complexas, é composta pelas modalidades de internação domiciliar e visita domiciliar.

A internação domiciliar, segundo Tavolari, Fernandes e Medina (2000), refere-se ao cuidado intensivo, contínuo e multiprofissional, desenvolvido em casa. Requer aparato tecnológico semelhante ao existente na estrutura hospitalar, pois o cuidado é direcionado aos doentes com complexidade moderada ou alta, no entanto, esses recursos tecnológicos são solicitados de acordo com a necessidade do cliente.

Pacientes com problemas de saúde moderados ou leves, necessitam da visita domiciliar. A qual é realizada por um profissional ou equipe para avaliar e estabelecer um plano assistencial de saúde. De acordo com Jacob (2001), as orientações realizadas neste tipo de atendimento, dizem respeito ao saneamento básico, cuidados com a saúde, uso de medicamentos, amamentação, controle de peso, ou qualquer coisa que diga respeito àquele indivíduo, à família e à comunidade em que vivem.

2.1.2. Internet das Coisas (IoT)

O termo *Internet* das Coisas nomina a revolução tecnológica marcada pela conexão dos objetos físicos usados no cotidiano à rede mundial de computadores. Essa tecnologia permeia a comunicação de máquinas entre si e de máquinas com pessoas. Os dados são reunidos e transmitidos por intermédio de *softwares* próprios e de sensores inteligentes.

É um novo paradigma computacional, sua utilização cresce significativamente e pode modificar a forma com que os seres humanos interagem com os computadores [Koreshoff et al. 2013]. Em consonância com isso, Davis (2018) estimou que em 2020 estarão conectados à *internet* cerca de 50 bilhões de dispositivos.

Tamilselvi et al. (2020) ressaltam a importância da *IoT* para a saúde. Conforme seu trabalho, para pacientes em coma o monitoramento contínuo dos parâmetros físicos pode manter até 60% das vidas humanas, através da detecção oportuna na variação dos sinais vitais dos pacientes.

2.1.3. Metodologia *Secure-SSDM*

A Metodologia de Desenvolvimento Solo de Software Seguro (do inglês, *Secure Solo Software Development Methodology (Secure-SSDM)*) foi elaborada para ser usada em projetos de *software* com apenas um desenvolvedor. Essa estrutura garante um processo ágil na construção de *softwares* seguros e de qualidade.

De acordo com Moyo e Mnkandla (2020), o *Secure-SSDM* é uma metodologia ágil planejada mediante a associação de práticas de qualidade e práticas de segurança retiradas da literatura. Isso atribui à ferramenta características intrínsecas de qualidade.

2.2 Hardwares

Os componentes eletrônicos presentes neste projeto serão identificados nas subseções seguintes.

2.2.1. Microcontrolador ESP32

Este microcontrolador foi desenvolvido especialmente para eletrônicos vestíveis, dispositivos móveis e aplicações *IoT*.

A Figura 1 apresenta o chip ESP32, o qual possui conectividade *Wi-Fi* e *Bluetooth* integradas. Este microcontrolador é capaz de funcionar em um ambiente industrial com temperaturas que podem variar de -40° C até 125° C.



Figura 1. Microcontrolador ESP32 [Espressif 2019]

2.2.2. Sensor de Temperatura LM35

O sensor LM35 é um sensor de precisão desenvolvido para medir a temperatura de objetos.

Possui tensão de saída de 10mV/ °C, ou seja, cada 10mV na saída representa um grau Celsius. A Figura 2 ilustra sensor de temperatura LM35, nela é possível identificar ao lado esquerdo, o pino do polo positivo, ao centro o pino de saída analógica e a direita, o pino do polo negativo.



Figura 2 Sensor de Temperatura LM35 [Robocore 2020]

2.2.3. Sensor de Batimento Cardíaco e Oxímetro Max30100

O sensor Max30100 é usado para medir os batimentos cardíacos e a saturação de oxigênio no sangue. Este sensor combina dois *led's* e um tubo de recepção.

Na figura 3a é possível visualizar o funcionamento do Max30100, aonde, após os *led's* emitirem feixe de luz contra a superfície da pele, o fotodiodo observa a variação na intensidade desta luz refletida.

Este sensor baseia-se no princípio da variação do volume de sangue. O sangue arterial absorve luz e quando a quantidade de sangue muda, resulta em alterações na intensidade da luz recebida pelo fotodiodo [Wan et al. 2017].

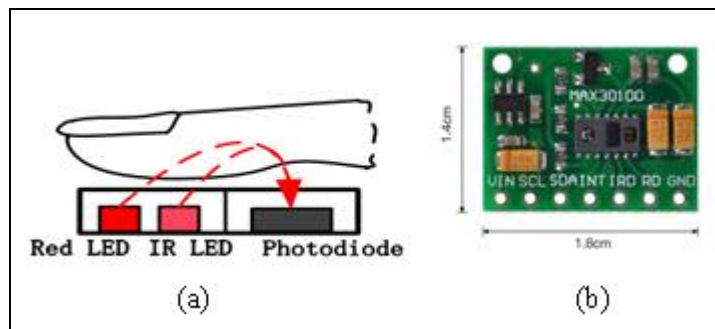


Figura 3. Sensor Max3010 [Wan et al. 2017]

O módulo, conforme ilustra a Figura 3b, opera na faixa de tensão entre 1.8 e 3.3 *volts*, o baixo consumo de energia permite seu funcionamento por longo tempo, mesmo conectado à bateria de baixa tensão.

2.3. Protocolo

Esta subseção apontará o protocolo utilizados na comunicação deste projeto.

2.3.1 Protocolo de Comunicação MQTT

O protocolo MQTT foi criado em 1999 pela IBM, com a finalidade de possibilitar troca de mensagens entre máquinas com poucos recursos de *hardware* e de rede.

A comunicação neste protocolo, acontece de forma assíncrona, sem necessitar de qualquer tipo de requisição, pois está sob o paradigma de *Publisher, Subscribe e Broker*. Os dispositivos do tipo *Publisher* são responsáveis por publicar as mensagens no *Broker*, o qual, administra o fluxo das publicações, enviando as mensagens para os dispositivos *Subscribe*, mas só recebe as informações os dispositivos que têm interesse na mensagem publicada, os demais não tem acesso.

Este protocolo é amplamente utilizado na *IoT*, especialmente em sistemas embarcados, pois o protocolo é leve e sua codificação é simples. Seu funcionamento será eficaz mesmo em redes de internet com largura de banda limitada e em *hardwares* de dispositivos restritos [Yuan 2017].

3. Trabalhos Correlatos

Nesta sessão serão apresentados os trabalhos relacionados ao proposto, envolvendo semelhanças na implementação e desenvolvimento de dispositivos com tecnologia *IoT* para monitoramento dos sinais vitais dos pacientes.

3.1 Protótipo da frequência cardíaca do grupo Monitorando com ESP32

Skraba et al. (2019) apresentou em seu trabalho o desenvolvimento de um sistema para monitoramento de frequência cardíaca em grupo. Baseado no módulo ESP32, este dispositivo pode ser usado em ambientes clínicos, atividades esportivas em grupo, experimentos sobre estresse em grupo e na vigilância sem fio de pessoas durante situações perigosas (militar, combate a incêndios).

Para monitorar a frequência cardíaca, os autores elaboraram um aparelho que conecta o sensor de pulso PPG à orelha do paciente. Esse sensor é capaz de detectar alterações no volume de sangue microvascular. Após a aquisição dos dados, o sensor envia essas informações ao microcontrolador ESP32 que se comunica com a *internet* por via Wi-Fi, e com celulares, tablets e TVs inteligentes por via *BlueTooth* (BT) ou *BlueTooth Low Energy* (BLE). Utiliza o protocolo *WebSocket* e servidor implementado em node.js para deixar as informações disponíveis na *Web*.

O trabalho citado tem como característica similar a este projeto o fato de utilizar o microcontrolador ESP32 para monitorar os sinais vitais das pessoas. Porém, há diferenças no que tange a quantidade e o tipo do sensor, e a principal divergência encontra-se no uso do *WebSocket*, que pode causar lentidão na comunicação devido à complexidade desse protocolo.

3.2 Medição de Sinais Vitais usando a Tecnologia *Internet das Coisas*

Kristiani et al. (2019) mostram em seu trabalho, a importância da medição dos sinais vitais para diagnosticar doenças e determinar tratamento médico apropriado. Este estudo objetiva projetar a frequência respiratória e monitorar a frequência cardíaca.

Os autores desenvolveram um aparelho para detectar a respiração usando um sensor flexível e para monitorar a frequência cardíaca foi usado o sensor SEN11574. Os resultados desses sensores são enviados à *internet* via microcontrolador ESP32. Os parâmetros foram mostrados na *Web* usando a *Internet das Coisas* (IoT), com base no aplicativo e API *ThingSpeak*. Em testes realizados, o

sistema demonstrou resultados com margem de erro de 2,3% para frequência respiratória e de 0,57% para a frequência cardíaca.

As semelhanças entre o trabalho correlato e o presente projeto, são evidentes pois ambos utilizam a tecnologia IoT para monitorar os parâmetros fisiológicos dos pacientes. Contudo, as principais diferenças são a escolha de quais sinais vitais serão analisados e quais os sensores serão empregados.

4. Proposta

Este projeto propõe uma forma de monitorar os sinais vitais do paciente à distância, extrair os dados e enviá-los à equipe de saúde para garantir a detecção imediata da necessidade de cuidados médicos específicos. Dessa forma, é possível determinar a intervenção médica apenas para os pacientes que apresentarem parâmetros vitais gravemente alterados.

A Figura 4 ilustra a proposta, a qual consiste na construção de um protótipo baseado na tecnologia *IoT*, para monitorar a frequência cardíaca, medir a saturação de oxigênio no sangue e mensurar a temperatura do paciente no atendimento domiciliar à saúde.

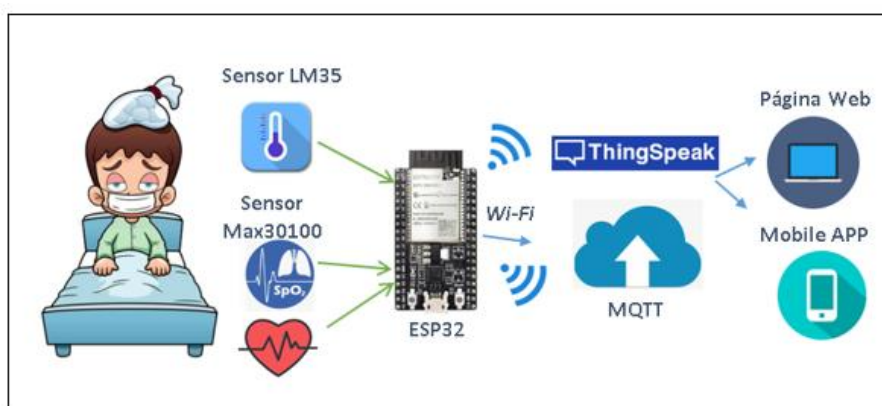


Figura 4. Proposta do Projeto

Na construção do *hardware*, serão usados o Sensor de Frequência Cardíaca Max30100 para medir a saturação de oxigênio no sangue e monitorar os batimentos do coração e o Sensor LM35 para mensurar a temperatura do doente. Os resultados desses parâmetros serão enviados ao microcontrolador ESP32, e processados através de comandos implementados na IDE em linguagens C/C++.

Em relação a elaboração da comunicação, será usado o microcontrolador ESP32 para a conexão com a *Internet* via Wi-Fi, e por meio do protocolo MQTT enviar os valores de cada parâmetro colhido para a plataforma *ThingSpeak*, permitindo a visualização dos dados na *web*. Além disso, é possível baixar um aplicativo da própria plataforma e visualizar as informações no celular.

Por conseguinte, a metodologia *Secure-SSDM* foi escolhida para o desenvolvimento deste projeto. Suas etapas serão discutidas no subtópico Projeto de *software*.

4.1. Projeto

Esta subseção tem o propósito de apresentar as etapas do projeto de *hardware* e de *software* da proposta.

4.1.1. Projeto de *hardware*

O projeto de *hardware* é dividido em duas partes: Diagrama Esquemático do *hardware* (a), e Estrutura física do dispositivo (b). A primeira, apresentando a ligação entre os componentes; e a segunda, ilustra a forma física que o dispositivo terá.

a. Diagrama Esquemático do *hardware*

Quanto a primeira parte da modelagem de hardware, a representação do projeto eletrônico desenvolvido foi construída com o auxílio do *software Fritzing*. A Figura 5 ilustra a ligação entre o microcontrolador (c) e os dois sensores (a, b).

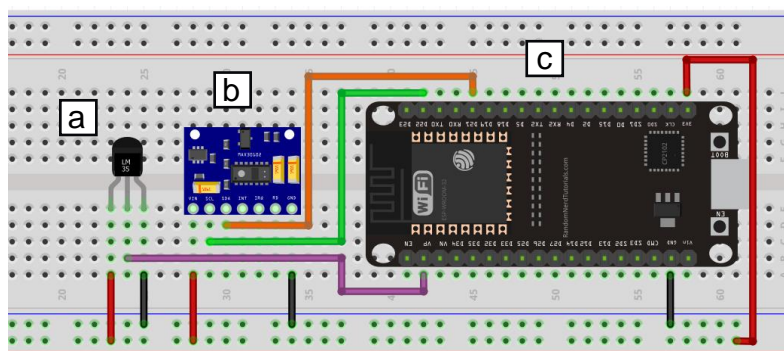


Figura 5. Esquemático do circuito de *hardware*

Na modelagem representada pela Figura 5, há dois sensores, o Max30100 (b), responsável por medir a frequência cardíaca e a saturação de oxigênio no sangue e o Lm35 (a), responsável por aferir a temperatura, ambos serão conectados ao microcontrolador ESP32 (c), onde encontra-se o algoritmo desenvolvido para a leitura dos sensores e envio de dados para a plataforma *ThingSpeak*.

b. Estrutura física do sistema

A representação da estrutura do protótipo, conforme Figura 6, foi construída utilizando o *software Sketchup Free*, que permite modelar elementos 3d em sua plataforma *web*.

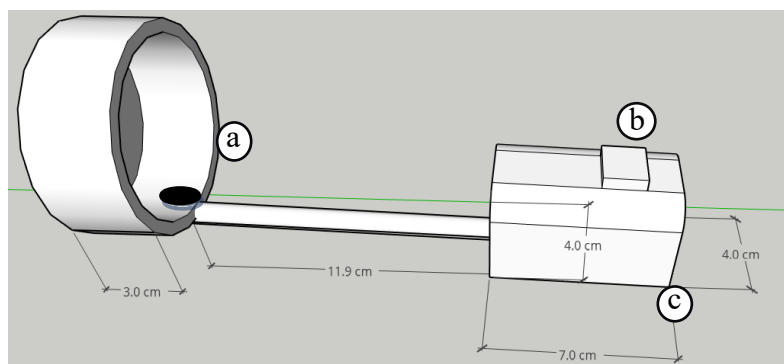


Figura 6. Projeto do protótipo do *hardware*

O desenho ilustrado na Figura 6, apresenta a pulseira contendo o sensor de temperatura LM35 (a) e a carcaça incluindo o ESP32 (c) e na superfície o sensor Max30100 (b), onde o paciente deverá posicionar seu dedo para obter o batimento cardíaco e a saturação de oxigênio no sangue.

4.1.2. Projeto de *Software*

Este projeto foi orientado pela metodologia *Secure-SSDM*, cujas principais definições de cada fase serão dispostas abaixo:

Fase I: Foi definido o uso das linguagens de programação C e C++ e o Arduino IDE como ambiente de desenvolvimento, além do uso da ferramenta de gerenciamento de tarefas *Trello* contendo as colunas *Product Backlog*, *a fazer*, *fazendo* e *concluído*.

Fase II: Inicialmente, elaborou-se as histórias do usuário (Apêndice B), os casos de uso (Apêndice C), além do diagrama de atividades (Apêndice D) e por fim gerou-se uma lista de tarefas;

Fase III: Nesta etapa, priorizou-se a lista de tarefas, criou-se o *Product Backlog* presente na Tabela 1; por fim, definiu-se o *Sprint Planning* das tarefas a serem realizadas no prazo máximo de 14 dias, tempo estipulado para cada *Sprint*.

Tabela 1. *Product Backlog* do Sistema de Monitoramento

ID	TÍTULO	ESTIMATIVA	PRIORIDADE
06	Conectar à <i>Internet</i>	34	Essencial
07	Visualizar os Sinais Vitais	21	Essencial
01	Medir a Frequência Cardíaca	13	Essencial
03	Medir a Saturação de Oxigênio no Sangue (SpO2)	13	Essencial
02	Medir a Temperatura	5	Essencial
04	Configurar Situação de Risco	8	Opcional
05	Receber Alerta de Situação de Risco	8	Opcional

Fase IV: Aqui, iniciará a *Sprint* com a produção de código fonte para as tarefas. Será utilizado a ferramenta *Git* para garantir o controle de versionamento.

Fase V: Neste estágio, acontecerá a revisão do código e o fechamento de cada *Sprint* do desenvolvimento. Sendo que, as tarefas concluídas são direcionadas à coluna *concluído*, as incompletas, retornam para compor a próxima *Sprint*.

Fase VI: Por fim, o produto criado foi avaliado em relação ao produto projetado.

5. Implementação

Nesta subseção está descrito o processo de construção do *hardware* e do *software* deste sistema.

5.1. Hardware

Para a fabricação do *hardware* foi confeccionado uma caixa de plástico ABS, com auxílio da impressão 3d baseada no projeto criado para a estrutura física. O circuito foi montado sob uma placa de fenolite onde foram soldados os fios que ligam os sensores ao microcontrolador. Esta placa, juntamente com o microcontrolador (b), foram acomodados dentro da caixa, e fixados com adesivo termoplástico.

O sensor Max30100 (c) foi instalada na parte superior da caixa, e fixado com o adesivo termoplástico, já o sensor LM35 (a) foi preso à pulseira, a qual foi confeccionada em tecido do tipo Tergal, por ser confortável e resistente.

Na figura 7, é apresentada a imagem do protótipo construído, juntamente com todos os componentes eletrônicos e a fonte de alimentação (d) do *hardware*.

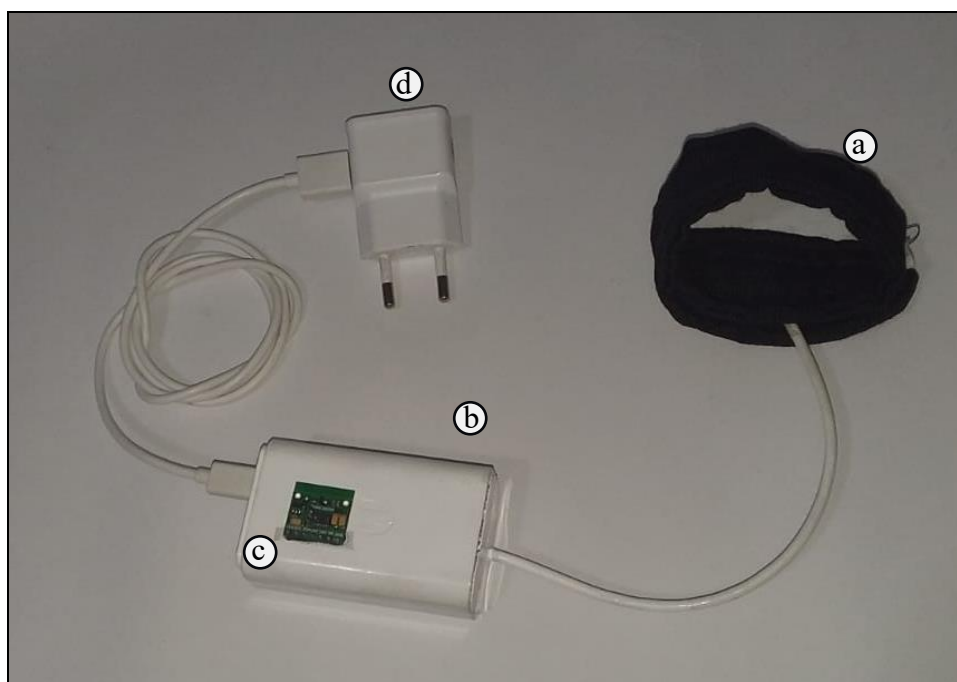


Figura 7. Protótipo do Sistema para monitoramento remoto

5.2. Software

Neste subtópico serão expostas as principais funcionalidade implementadas neste sistema: A aquisição da temperatura realizada pelo sensor LM35, a medição da frequência cardíaca e da saturação de oxigênio no sangue aferidas pelo sensor Max30100; A comunicação do microcontrolador com a *internet*; O envio dos dados sob o protocolo MQTT e a apresentação dos dados na plataforma *ThingSpeak*.

5.2.1. Aquisição dos Dados

A leitura dos sensores foi realizada com um intervalo de 5 segundos entre cada leitura.

Para adquirir a temperatura, a frequência cardíaca e a saturação de oxigênio no sangue, é de vital importância para o projeto. Portanto, primeiramente é obtido o valor da temperatura, através da função “*getTemperaturaCelsius()*”.

```

102 float milivolts = analogRead(LM35) * (3300/1024);
103 return (milivolts/10);

```

Figura 8. Trecho do código aquisição da temperatura

A Figura 8 apresenta a fórmula para se obter a temperatura em graus *Celsius*. Para tanto, o ESP32 recebe do Sensor LM35 um valor de 0 até 1023, onde 0 representa 0 *volt* o valor 1023 equivale a 3,3 *volts*, então para saber a tensão que o sensor está enviando, basta efetuar regra de três simples: se 3.3 *volts* equivale a 1023, então o valor do sensor em *volts* é igual ao valor analógico do sensor vezes a 3,3 /1024. O valor resultante é dado em *volts*, então é preciso convertê-lo para *milivolts*, multiplicando-o por mil. E para se obter a temperatura, deve-se dividir por 10, pois a cada 10 *milivolts* equivale a um grau *Celsius*.

Em seguida, é adquirido os valores da saturação de oxigênio no sangue e da frequência cardíaca. Para esse fim, é utilizada as funções pré-definidas da biblioteca “MAX30100_PulseOximeter.h”, a qual possibilita a comunicação entre o sensor e o microcontrolador. Ainda há métodos para conexão e leitura do sensor.

Para a utilização desta biblioteca, é necessário instanciar um objeto da classe “PulseOximeter”, chamado de “pox”. O método “begin()” inicializa esta classe. A Figura 9 apresenta o método “begin()” executado na função “setup()” da IDE.

```

64     if (!pox.begin()) {
65         Serial.println("[MAX30100]--FAILED");
66         for(;;);
67     } else {
68         Serial.println("[MAX30100]--SUCCESS");
69     }

```

Figura 9. Inicialização da conexão do sensor MAX30100

Ainda na função “setup()”, é possível ajustar o valor da corrente no sensor MAX30100, utilizando a função “pox.setIRLedCurrent()”, que recebe o parâmetro contendo a corrente desejada, que por padrão é de 50mA.

Este projeto utilizará 7,6mA, conforme a seguinte sintaxe: “pox.setIRLedCurrent(MAX30100_LED_CURR_7_6MA)”.

Em seguida, na função “loop()”, realiza-se a aquisição dos dados. Para esse fim, são utilizados os métodos “getSpO2()” e “getHeartRate()” contidos igualmente na biblioteca “MAX30100_PulseOximeter.h”. O primeiro, retorna o valor da saturação de oxigênio no sangue, e o outro, resulta no valor da frequência cardíaca. A Figura 10 apresenta os métodos utilizados.

```

89     SpO2 = pox.getSpO2();
90     frequenciaCardiaca = pox.getHeartRate();

```

Figura 10. Métodos getHeartRate () e getSpO2()

5.2.2. Comunicação com a Internet

A comunicação com a *internet* será efetuada através da rede *WiFi*, com auxílio da biblioteca “*WiFi.h*”. Com ela é possível inicializar a conexão, executando a função “*WiFi.begin()*”, passando dois parâmetros: o nome da rede e a senha, e para cada tentativa de conexão, é impresso um ponto na porta serial. Para saber o *status* da conexão, usa-se o método “*WiFi.Status()*”, caso retorne a propriedade “*WL_CONNECTED*”, o dispositivo está conectado, caso contrário é necessário reconectar. A Figura 11 apresenta a utilização da biblioteca “*WiFi.h*”.

```
123 while (WiFi.status() != WL_CONNECTED){
124     WiFi.begin("nome_da_rede", "senha");
125     Serial.print(".");
126     delay(2500);
```

Figura 11. Funções *WiFi.begin()* e *WiFi.status()*;

5.2.3. Envio dos Dados

Os dados serão enviados a cada 5 minutos para a plataforma *ThingSpeak*, esta comunicação acontecerá sob o protocolo MQTT, onde o microcontrolador *publicará* seus dados no MQTT *broker* desta plataforma. Para isso, é necessário utilizar a biblioteca “*PubSubClient.h*”. A inicialização desta biblioteca é apresentada conforme a Figura 12.

```
35 WiFiClient client ;
36 PubSubClient MQTT(client);
```

Figura 12. Inicialização da Biblioteca *PubSubClient.h*

Após os objetos instanciados, apresentados na Figura 12, é preciso configurar o servidor, passando como parâmetros o endereço e porta do servidor, através do método “*setServer()*”. Ainda é possível configurar a função de *callback*, pelo método “*setCallback()*”. A Figura 13 apresenta os métodos “*setServer()*” e “*setCallback()*”.

```
137 MQTT.setServer("mqtt.thingspeak.com", 1883);
138 MQTT.setCallback(callback);
```

Figura 13. Métodos *setServer()* e *setCallback()*

Além disso, a conexão com o *broker* fica por conta do método “*connect()*”. Já a envio dos dados ao *broker* é realizada pelo método “*publish()*”, o qual recebe dois parâmetros: *topic* e *dados*. A variável *topic* é apresentada na Figura 14, onde é constituída de um “*ChannelID*” e “*WriteAPIKey*”, cujos valores são informados pela plataforma *Thingspeak*.

```
23 //char* topic="channels/<channelID/publish/<channelAPI>
24 char* topic = "channels/IDCanal/publish/MXVR6TAS36NYHC5B";
```

Figura 14. Variável *topic*

Dessa forma, a constituição da variável *dados* e a utilização do método “*publish()*” são apresentados na Figura 15.

```
String dados = "field1=";
dados += fCardiaca;
dados += "&field2=";
dados += spO2;
dados += "&field3=";
dados += temp;
dados += "&status=MQTTUBLISH";
if (MQTT.publish(topic, (char*) dados.c_str())) {
    Serial.println("Publish [ok]");
} else {
    Serial.println("No Publish [FALHA]");
}
```

Figura 15. Variável *dados* e método *publish()*

A formação da variável *dados*, conforme Figura 15, é composta dos elementos “*field1*”, “*field2*”, “*field3*” para demarcar em que campo da plataforma os dados serão apresentados.

5.2.4. apresentação das informações na plataforma *ThingSpeak*.

Os profissionais de saúde necessitaram utilizar a plataforma *ThingSpeak* para acompanhar os sinais vitais dos paciente. Esta plataforma permite acompanhar de forma dinâmica a alteração a alteração nos dados adquiridos pelos sensores e enviados à plataforma pelo microcontrolador. Toda vez que o sistema envia os valores dos sensores, a plataforma atualizará automaticamente o gráfico, plotando os novos pontos no gráfico com os valores atualizados.

Na Figura 16 é possível visualizar os gráficos da temperatura, da frequência cardíaca e da saturação de oxigênio no sangue

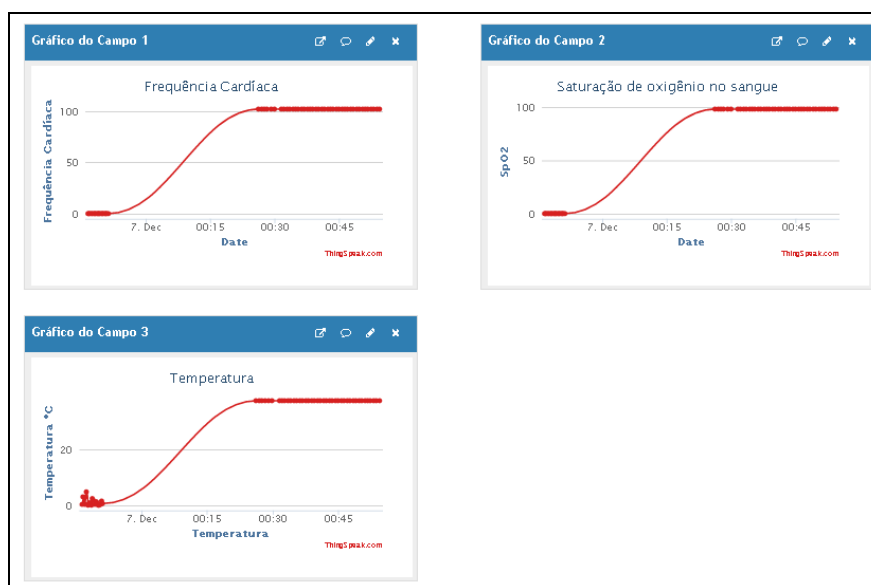


Figura 16. Gráficos ThingSpeak [ThingSpeak, 2020]

6. Cenário de testes e validação

Os testes de validação foram realizados por meio de comparação de valores, para tanto, os valores das temperaturas medidas no sistema foram comparadas aos valores aferidos em um termômetro analógico de mercúrio. Já os testes referentes aos batimento cardíaco e à saturação de oxigênio no sangue aconteceu por meio da comparação dos valores aferidos no sistema com os valores medidos no Oxímetro Digital de Dedo, da marca Multilaser HC261.

Deste modo, efetuou-se dez leituras de cada sinal vital em seis pessoas, e efetuou-se a média aritmética destes valores. As faixas etárias dos participantes variam de 4 até 66 anos e de ambos os sexos. A Tabela 2, apresenta um comparativo formada pelas seguintes colunas: Identificação dos participantes quanto ao sexo (S) e idade (I), T (temperatura) medida em Graus Celsius, FC (frequência cardíaca) aferida em batimentos por minuto e SpO2 (saturação de oxigênio no sangue) em porcentagem (%). Foram realizadas as 10 leituras intercalando entre o sistema desenvolvido e os outros aparelhos.

Tabela 2. Comparativo entre os dispositivos

Identificação		Dados do Dispositivo			Outros Equipamentos			Diferença de valores		
S	I	T	FC	SpO2	T	FC	SpO2	T	FC	SpO2
F	4	44,3	72	95	36,2	87	97	8,1	15	2
M	7	47,1	82,7	96	36,8	93	98	10,30	10,3	2
M	15	49,8	84,3	97	35,6	84,2	96,3	14,20	0,01	0,7
F	34	45,9	88,34	96	36,4	89	97	9,5	0,66	1
F	59	49,8	88,7	96	35,2	90	95	14,6	1,3	1
F	66	47,1	87	97	35,6	89,8	97	11,15	2,8	0

7. Conclusão

O protótipo desenvolvido neste trabalho permitiu monitorar a frequência cardíaca, medir a saturação de oxigênio no sangue e mensurar a temperatura do paciente à distância.

As pesquisas realizadas sobre *IoT* aplicada à saúde, sobre a aquisição de valores em sensores LM35 e MAX30100, além do estudo realizado sobre a comunicação entre o protótipo e a plataforma *ThingSpeak* através do protocolo MQTT, possibilitaram realizar teste para validação deste sistema.

Um dos problemas encontrados durante o desenvolvimento deste trabalho, foi realizar a leitura do sensor LM35, que conforme Tabela 2, apresentou valores imprecisos na aferição da temperatura, percebeu-se que este não é indicado para medir a temperatura exata de pacientes.

Outra dificuldade observada foi na leitura do sensor MAX30100, que ao medir a frequência cardíaca apresentou valores duvidosos, principalmente quando o usuário movimentava o dedo sobre o sensor.

Mesmo com resultados desqualificando o uso deste protótipo para monitorar a temperatura e a frequência cardíaca, este trabalho contribui para o desenvolvimento de novas soluções utilizando a *internet* das coisas aplicada à saúde, em razão de que apresentou o processo de criação deste sistema, evidenciando os acertos e apontando os erros a serem corrigidos.

7.2 Trabalho Futuros

Uma das sugestões propostas será melhorar o *design* do dispositivo, dando-lhe um formato anatômico, proporcionando conforto ao paciente e garantindo melhor performance dos sensores. Outra sugestão será a implantação da *Situação de Risco e Alerta de risco*. Assim, será possível atribuir a cada sinal vital um valor tido como arriscado para a saúde do paciente. Caso este valor seja alcançado, automaticamente será enviada uma mensagem ao responsável informando o alerta de risco.

Referências

- Arduino. (2020) “Open-source electronics prototyping platform”. Acesso em maio de 2020. Disponível em: <<http://www.arduino.cc>>.
- Barro, S. et al. (2002) “Novas tendências no monitoramento de pacientes.”, In: Gerenciamento de doenças e resultados em saúde, vol.10, nº 5, p. 291-306.
- Beas, J. M. (2011) “Histórias de usuário”. Acesso em maio de 2020. Disponível em: <<https://jmbeas.es/guias/historias-de-usuario/>>
- Bernabé, R. B; Navia, I. A; Garcpia-Peñalvo, F. J. (2015) “Faat: freelancer em equipe”. In: Anais da 3ª Conferência Internacional sobre Ecosistemas Tecnológicos para Melhorar a Multiculturalidade. p. 687-694.
- BRASIL. Ministério da Saúde. Agência Nacional de Vigilância Sanitária. Resolução RDC nº 11, de 26 de janeiro de 2006. Diário Oficial da União, Poder Executivo, Brasília, DF, 30 jan. 2006. Acesso em: maio 2020 Disponível em: <<http://e-legis.anvisa.gov.br/leisref/public/showAct.php?id=20642&word=rdc%202006%20domiciliar>>.
- Casa da Robótica. (2019) “Sensor Frequência Cardíaca Max30100 Oxímetro”. Acesso em maio de 2020. Disponível em: <<https://www.casadarobotica.com/sensores-modulos/sensores/outros/sensor-frequencia-cardiaca-max30100-oximetro>>
- Crosby, P.B. (1979) “Quality is free”. New York: McGraw-Hill.
- Espressif. (2019) “DevKits da série ESP32”. Acesso em maio de 2020. Disponível em: <<https://www.espressif.com/en/products/devkits>>
- Davis, G (2018), "2020: Life with 50 bilhões de dispositivos conectados", 2018 IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, NV, 2018, pp. 1-1, doi: 10.1109 / ICCE.2018.8326056.
- Duarte, Y. A. O.; Diogo, M. J. E. (2000) “Atendimento domiciliar: um enfoque gerontológico”. São Paulo: Atheneu.
- Gartner, R. C. (2017). Gartner says 8.4 billion connected “things” will be in use in 2017.

- Jacob, W. (2001) “Digerir as diferenças”. *Médicis*, São Paulo, n. 13, p. 3-8, nov./dez.
- Koreshoff, T. L., Robertson, T., and Leong, T. W. (2013). Internet of things: A review of literature and products. In *Proceedings of the 25th Australian Computer-Human Interaction Conference: Augmentation, Application, Innovation, Collaboration, OzCHI'13*, pages 335–344, New York, NY, USA. ACM.
- Lacerda, M. R, et al. (2006) “Atenção à saúde no domicílio: modalidades que fundamentam sua prática”. *Saúde e sociedade*, 2006, 15: 88-95.
- Moyo, S; Mnkandla, E. (2020) “A Novel Lightweight Solo Software Development Methodology With Optimum Security Practices”. *IEEE Access*, 2020, 8: 33735-33747.
- Organização Mundial de Saude. (2020) “Pandemia da doença de Coronavírus (COVID 19)”. Acesso em junho de 2020. Disponível em: <https://www.who.int/emergencies/diseases/novel-coronavirus-2019?gclid=CjwKCAjw26H3BRB2EiwAy32zhYwi5QyqUrl3VIsb93SPp7-aK26Y2F1VW6bzv7iyV00a2IUljEmuYBoCfjEQAvD_BwE>
- Potter AP, Perry A. (2009) “Fundamentos de Enfermagem”. 7ª edição, Brasil, Elsevier Editora Ltda, p. 503.
- Robocore. (2020) “Sensor de Temperatura – LM35”. Acesso em maio de 2020. Disponível em: <<https://www.robocore.net/loja/sensores/sensor-de-temperatura-lm35>>
- Tavolari, C. E. L.; Fernandes, F.; Medina, P. (2000) “O desenvolvimento do 'Home Health Care' no Brasil”. *Revista de Administração em Saúde*, São Paulo, v. 3, n. 9, p. 15-18, out./dez. 2000.
- ThingSpeak. (2020) “ThingSpeak para projetos de IoT”. Acesso em dezembro 2020. Disponível em: <<https://www.thingspeak.com/>>.
- Wan, J. et al. (2017) “Sistema de detecção de saturação de oxigênio no sangue tipo reflexivo baseado no MAX30100”. In: *2017 Conferência Internacional sobre Segurança, Análise de Padrões e Cibernética (SPAC)*. IEEE, 2017. p. 615-619.
- Yuan, Michael. (2017) “Conhecendo o MQTT”. Acesso em maio 2020. Disponível em: <https://developer.ibm.com/br/technologies/iot/articles/iot-mqtt-why-good-for-iot/>

Apêndice A – Metodologia *Secure-SSDM*

A Figura 1 apresenta o fluxo do processo desta metodologia, e identifica as ações a serem em cada uma das 6 etapas.

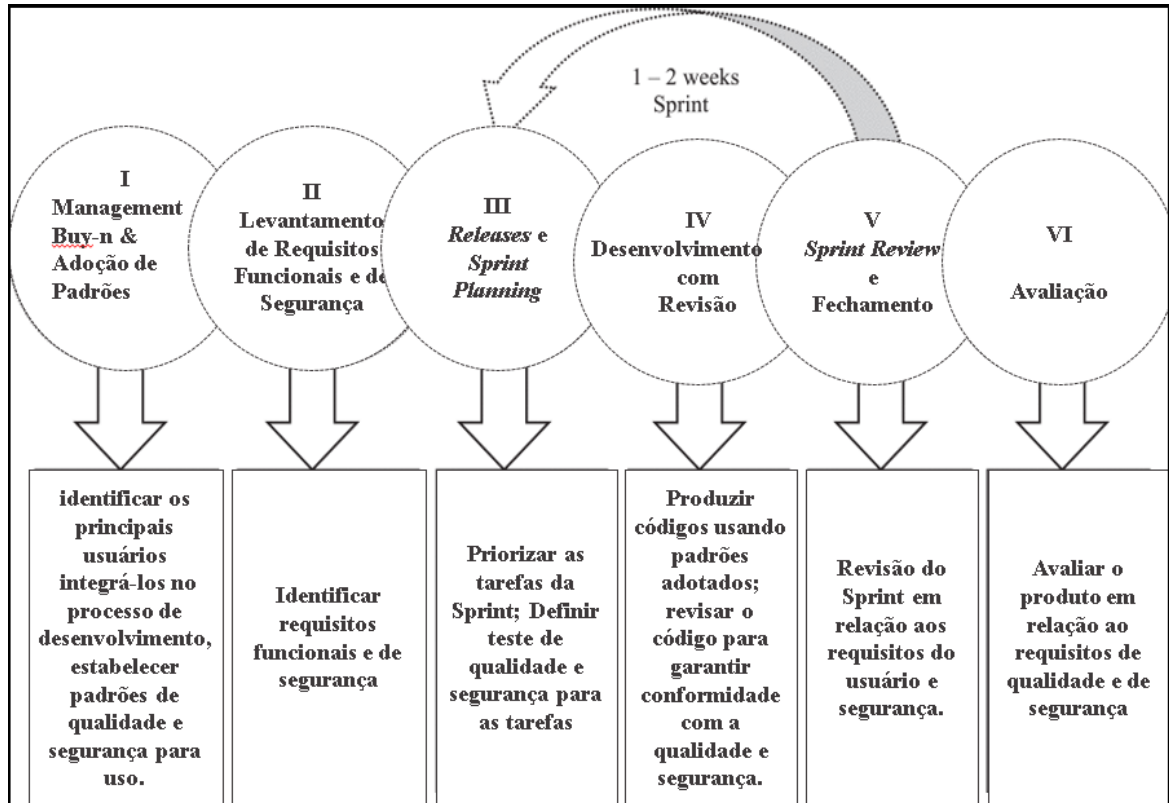


Figura 17. Estágios Secure-SSDM e principais atividades em cada estágio [Moyo e Mnkandla 2020]

Apêndice B. Levantamento de Requisitos Funcionais e de Segurança

O levantamento de requisitos do sistema foi feito através da coleta de histórias do usuário. Conforme Beas (2011), “Uma história de usuário descreve a funcionalidade que, por si só, agrega valor ao usuário”. Cada história do usuário descreve um requisito conforme Tabela 5.

Tabela 3. Histórias do usuário

ID	Título	Prioridade	Tema
	[História do usuário]		[Forma(s) de Confirmação e/ou Critérios de Aceitação]
01	Medir a Frequência Cardíaca	Essencial	Aferição
	Como usuário preciso medir a frequência cardíaca do paciente, para avaliar, monitorar e identificar possíveis arritmias cardíacas provenientes de doenças como, por exemplo, o infarto.		O sistema deverá medir a frequência cardíaca no formato Valor bpm, onde o valor é um número inteiro e bpm é a unidade de medida batimentos por minuto.
02	Medir a Temperatura	Essencial	Aferição
	Como usuário preciso medir a temperatura corporal do paciente, para avaliar, monitorar e identificar doenças ou distúrbios imunológicos.		O sistema deverá medir o valor da temperatura em Graus <i>Celsius</i> .
03	Medir a Concentração de Oxigênio no Sangue (SpO2)	Essencial	Aferição
	Como usuário, preciso mensurar a concentração de oxigênio, para avaliar, monitorar e indicar doenças pulmonares.		O sistema deve medir o valor da saturação de oxigênio no sangue em porcentagem.
04	Configurar a Situação de Risco	Opcional	Aferição
	Como administrador, quero configurar um valor para cada sinal vital caracterizando Situação de risco.		O sistema permitirá apenas ao administrador configurar um valor para cada sinal vital, estipulado como situação de risco
05	Receber Alerta de Situação de Risco	Opcional	Comunicação
	Como médico, quero ser informado sobre situação de risco, para adotar rapidamente as medidas adequadas.		Assim que o dispositivo detectar alteração nos sinais vitais conforme valor configurado na Situação de risco, enviará mensagem ao médico responsável.
06	Conectar à Internet	Essencial	Comunicação
	Como usuário, quero que o dispositivo conecte se à <i>internet</i> , para enviar os dados coletados à <i>web</i> .		O dispositivo enviará os dados dos sinais vitais coletados através da rede <i>Wifi</i> e ficaram disponíveis na <i>web</i> .
07	Visualizar os Sinais vitais	Essencial	Comunicação
	Como usuário, que visualizar os sinais vitais do paciente.		O sistema permitirá ao usuário acessar os sinais vitais coletados por meio de site na <i>web</i> .

Apêndice C - Diagrama de caso de uso

A metodologia *Secure-SSDM*, na segunda etapa, sugere a utilização de Diagramas UML para confecção, a Figura 22 ilustra o Diagrama de caso de uso deste trabalho.

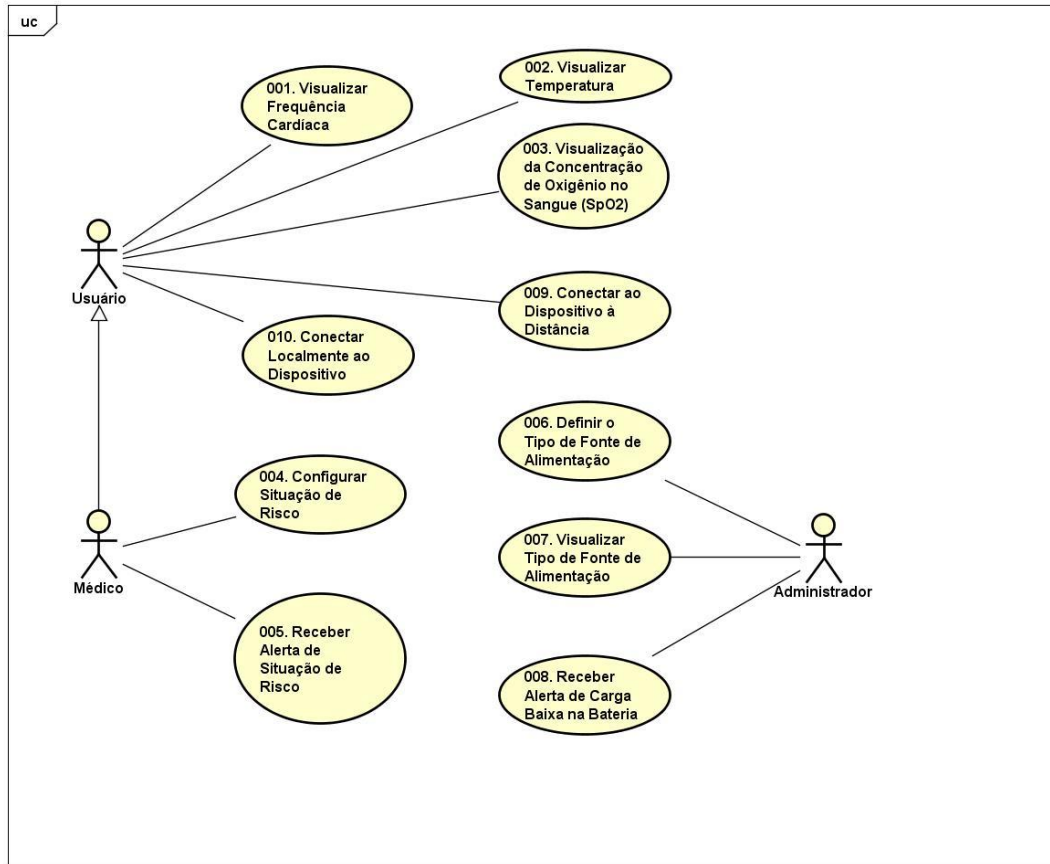


Figura 18. Diagrama de Caso de Uso

Apêndice D – Diagrama de Atividades

Ainda na Fase 2, a metodologia sugere a construção de um diagrama de atividade, para elucidar os requisitos do sistema.

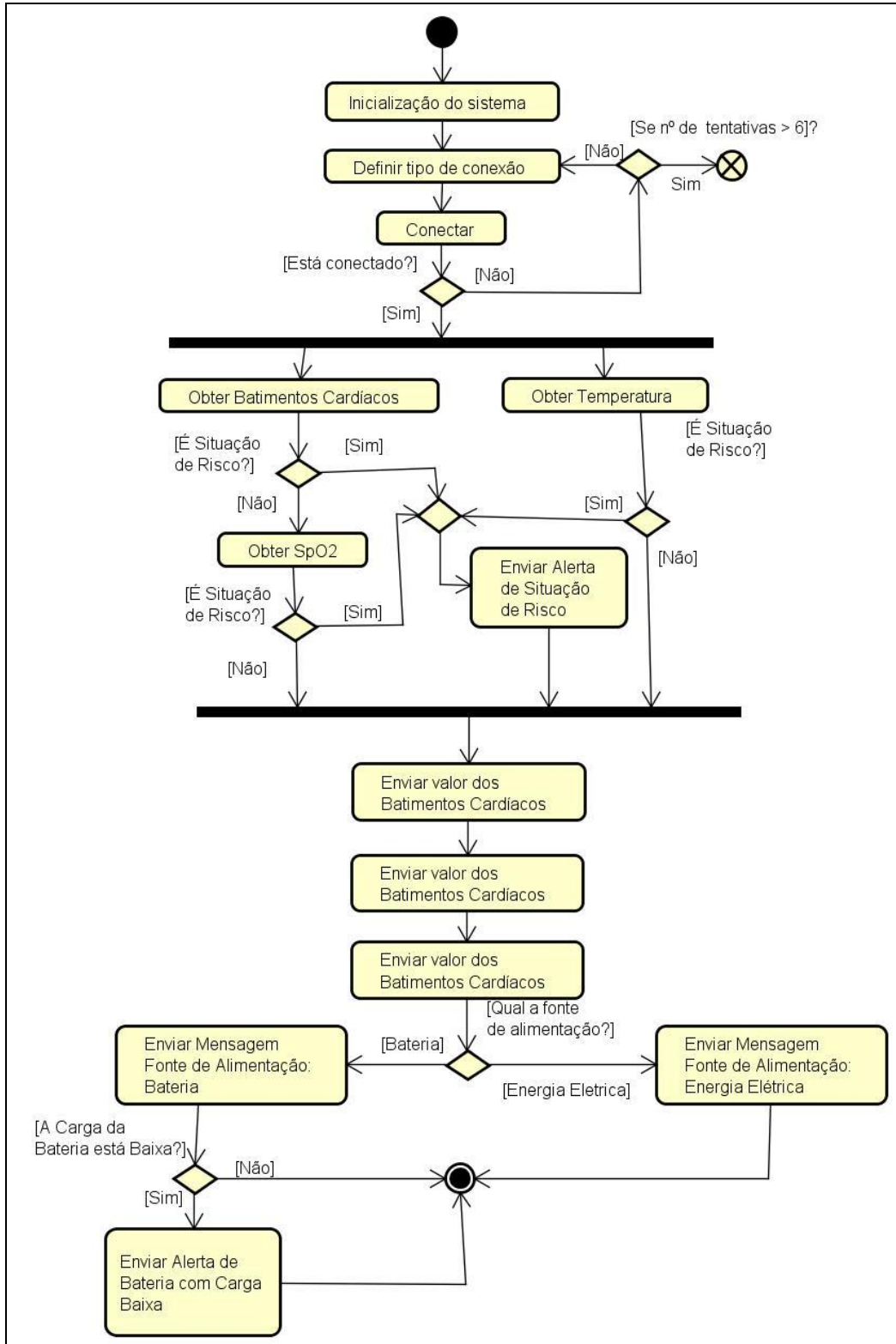


Figura 19. Diagrama de Atividade