

# Protótipo de detector de estresse

Luiza Castilho Ereno<sup>1</sup>, Helder Henrique da Silva<sup>1</sup>

<sup>1</sup>Centro de Ciências, Tecnologia e Saúde – Universidade Federal de Santa Catarina (UFSC)  
Araranguá – SC – Brazil

luiza.castilho@grad.ufsc.br, helder.h.silva@grad.ufsc.br

**Abstract.** *In the context of the pandemic, several studies highlighted the high level of stress in medical professionals and advised continuous monitoring of stress in these professionals. Thus, this work aims to develop a wearable prototype for detecting stress, contributing to the continuous and effective monitoring of these professionals in their work environments.*

**Resumo.** *No contexto da pandemia, diversos estudos ressaltaram o alto índice de estresse em profissionais da área médica e aconselharam o monitoramento contínuo de estresse nesses profissionais. Assim, esse trabalho tem como objetivo desenvolver um protótipo vestível para a detecção de estresse, contribuindo para o monitoramento contínuo e eficaz desses profissionais em seus ambientes de trabalho.*

## 1. Introdução

Internet of Things é o estado da internet caracterizado por dispositivos como sensores conectados a rede capazes de coletar dados, analisá-los e tomar decisões. A aplicação da Internet of Things na área médica é chamada de Internet of Medical Things, a IoMT marca a fase de monitoramento ubíquo de pacientes visando a detecção inicial de doenças e aplicação de planos de tratamento pró-ativos [Qadri et al. 2020].

A importância da integração dessas duas áreas se tornou ainda mais visível durante a pandemia, com os hospitais lotados e sem capacidade o suficiente para lidar com o vírus SARS-Cov-2, exames de rotina, como check up regulares foram adiados, atrasando o diagnóstico de pacientes em estágios iniciais.

Além dos pacientes, a equipe médica também teve sua rotina intensamente alterada. Segundo [Osório et al. 2021], um estudo conduzido com profissionais da saúde durante a pandemia, mais de 30% dos profissionais apresentaram problemas de saúde mental, como insônia, um indicativo perigoso, que causa piora no comportamento neurológico, colocando em risco não só o profissional como o paciente atendido por ele.

A revisão proposta por [Benfante et al. 2020] aponta que os profissionais de saúde tiveram sua saúde mental fortemente impactada pela pandemia, com taxa de sintomas clínicos relacionados a trauma entre 7,4 a 35%. A preocupação com a transmissão do vírus, a própria saúde, saúde da família, mudança no ambiente de trabalho e danos morais por ações, ou impossibilidade de implementar ações, que violam seus códigos de ética são fontes de estresse que aumentam as chances de transtornos mentais como abuso de drogas, depressão severa e transtorno pós-traumático. Assim, os autores destacam a importância da medição contínua de estresse na equipe médica, para diagnóstico em fases iniciais e tomadas de decisão para tratamento psicológico.

Vestíveis que coletam dados para o mapeamento de estresse já estão disponíveis no mercado em uma variedade de produtos. O estudo de [Chalabianloo et al. 2022] faz o levantamento de sete vestíveis, e obteve resultados satisfatórios de medição de estresse em todos eles. Porém, nem todos são comercializados no Brasil e todos apresentam custos elevados, o que inviabiliza o uso para a medição de estresse ocupacional para equipes médicas.

Desse modo, esse trabalho tem como objetivo desenvolver um protótipo de sistema IoT para medição e coleta de dados, de parâmetros relacionados ao estresse de um indivíduo, realizados por sensores e biomarcadores de fácil acessibilidade no mercado.

O presente artigo está estruturado da seguinte maneira: a segunda seção é referente a revisão de literatura, a descrição funcional do protótipo está na terceira seção, a quarta seção se refere à descrição estrutural, a quinta seção é referente ao desenvolvimento do protótipo e seus resultados e, por fim, na sexta seção é realizada a conclusão do trabalho.

## **2. Análise de Literatura**

O artigo [Arora and Meumeu Yomsi 2019] propõe um protótipo para monitorar sinais cardíacos de pessoas que vivem distante da cidade. Com os vestíveis e IOT, os pacientes não precisam se deslocar até a cidade, e podem ter seus biosinais medidos a qualquer momento, dentro de suas casas e em suas fazendas. Cinco sinais foram coletados: taxa de batimento cardíaco, pressão arterial sistólica, pressão arterial diastólica, saturação de oxigênio no sangue e temperatura corporal. Os dados são armazenados e podem ser consultados via site e aplicativo, e pela sensibilidade das informações, é necessário um login para entrar e ter acesso aos dados.

Já [Cholis et al. 2021] propõe um sensor que alerta para níveis iniciais de estresse usando fusão dos sensores que captam os seguintes parâmetros: frequência cardíaca, saturação de oxigênio, temperatura corporal, e resposta galvânica da pele. A fusão de vários sensores tem como objetivo aumentar a precisão. Para um monitoramento satisfatório, foram usados um aplicativo mobile e armazenamento automático em nuvem.

Em [Singh et al. 2023], o artigo propõe um sensor GSR, de atividade elétrica da pele, conectado a um Arduino, e posteriormente a um aplicativo mobile via módulo Bluetooth. LabVIEW foi integrado ao Arduino para a aquisição de dados. Para a calibração do sensor, 10 participantes foram expostos a 3 atividades: leitura, olhar um relógio e assistir vídeos, e o MATLAB foi usado para identificar os valores e classificar em 3 níveis de estresse: relaxado, normal e estressado.

O artigo [Singh et al. 2021] é uma continuação do último trabalho, com a criação de duas novas camadas: uma camada de comunicação intra-nodos, um nodo mestre que comanda até 7 nodos e é responsável por enviar os dados coletados via wifi para a nuvem, a segunda nova camada.

Em [Villarejo et al. 2012], o sensor GSR envia os dados coletados para um coordenador via ZigBee e esse envia para o computador. O coordenador era responsável por tomar ações para abaixar o nível de estresse do usuário, caso estivesse alto, como diminuir as luzes. Um problema relatado é que o dispositivo não consegue diferenciar uma ação forçada, como se forçar a rir, de estresse.

O artigo de [Gogate and Bakal 2019] usa o sensor GSR para criar um medidor

de estresse e fome para pacientes com capacidade reduzida de comunicação, mas com dificuldade em classificar fome e estresse simultaneamente já que quando o usuário está estressado, os sinais de fome são insignificantes.

Para melhor visualização, a tabela de extração dos protótipos achados nas bases de dados pode ser visualizada a seguir:

Referência	Sensores	Localização	Transmissão	Microcontrolador
[Singh et al. 2021]	GSR	Dedos	Bluetooth e internet	não mencionado
[Cholis et al. 2021]	GSR, TEMP, FC e SO	Dedos	Bluetooth e internet	não mencionado
[Singh et al. 2023]	GSR	Dedos	Bluetooth	Arduino e módulo bluetooth
[Arora and Meumeu Yomsi 2019]	PA, TEMP, SO e FC	Corpo	Internet	Arduino NANO e ESP8266
[Villarejo et al. 2012]	GSR	Dedo	ZigBee	
[Gogate and Bakal 2019]	GSR	Dedo	Internet	Arduino Nano e ESP8266

Tabela 1. Trabalhos Relacionados

### 3. Descrição Funcional

Para o desenvolvimento do sensor de estresse, o levantamento de requisitos funcionais e não funcionais foi realizado, eles especificam as funções que o protótipo deve ser capaz de fazer. Além disso, também foram especificadas as regras de negócio, que definem o comportamento do protótipo.

#### Requisitos Funcionais

Os requisitos funcionais do sensor de estresse incluem:

ID	Descrição
RF01	O sistema deve enviar notificações para o usuário se os níveis de estresse excederem um determinado limite.
RF02	O sistema deve proporcionar um dashboard, mostrando o nível atual de estresse.
RF03	O sistema deve guardar um histórico de dados coletados.
RF04	O sistema deve realizar coleta de dados de forma contínua.

Tabela 2. Requisitos Funcionais

#### Requisitos Não Funcionais

Os requisitos não funcionais do sensor de estresse incluem:

ID	Descrição
RNF01	O sistema deve garantir a privacidade e segurança dos dados do usuário.
RNF02	O aplicativo deve ser compatível com a web.
RNF03	O aplicativo deve ter um design responsivo, adaptando-se a diferentes tamanhos de tela.
RNF04	O aplicativo deve ser intuitivo, com uma curva de aprendizado de no máximo 10 minutos para novos usuários.

Tabela 3. Requisitos Não Funcionais

ID	Descrição
RN01	Os níveis de estresse são categorizados em "Baixo", "Moderado" e "Alto" com base nos dados coletados pelo hardware.
RN02	O feedback ao usuário deve ser baseado nas melhores práticas médicas e pesquisas atualizadas.
RN03	Cada usuário deve ter uma identificação única no sistema.

**Tabela 4. Regras de Negócio**

### Regras de Negócio

As regras de negócio do sensor de estresse incluem:

### Requisitos de Hardware

Os requisitos de hardware do sensor de estresse incluem:

ID	Descrição
RH01	O dispositivo de hardware deve ser capaz de se conectar de forma confiável à internet, preferencialmente usando Wi-Fi (ESP32).
RH02	O dispositivo deve incluir um sensor GSR, frequência cardíaca e temperatura para calcular o estresse do usuário.
RH03	O hardware deve possuir uma fonte de alimentação confiável, seja através de uma bateria recarregável ou conexão direta à energia elétrica.
RH04	O hardware deve transmitir dados ao servidor em intervalos regulares.
RH05	O dispositivo deve ter uma vida útil duradoura e ser resistente a desgastes comuns do dia-a-dia.

**Tabela 5. Requisitos de Hardware**

## 3.1. Gerenciamento do banco de dados do servidor

Essa seção se dedica a explicação do gerenciamento dos dados no servidor.

### 3.1.1. Arquitetura do Servidor

A arquitetura do servidor é fundamentada no uso do Django, um framework de aplicação web de alto nível escrito em Python que promove desenvolvimento rápido e design limpo. O Django foi selecionado devido à sua arquitetura MVT (Model-View-Template), que facilita a manutenção do código e a separação de lógica de negócios da apresentação da interface do usuário. Para fins de desenvolvimento e testes, o servidor Django foi configurado para operar localmente. Isso oferece a vantagem de um ciclo de desenvolvimento mais rápido, permitindo que os desenvolvedores façam ajustes imediatos e testem funcionalidades em tempo real sem a necessidade de implantação em um servidor externo.

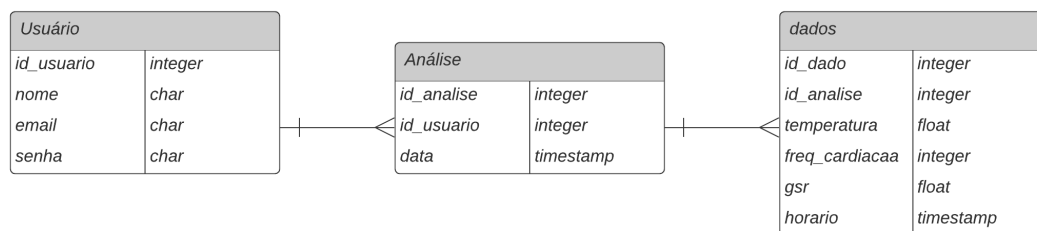
### 3.1.2. Modelagem do Banco de Dados

A modelagem do banco de dados foi projetada para suportar o armazenamento e a recuperação eficientes dos dados do usuário e das análises. Utilizando o MySQL, um

sistema de gerenciamento de banco de dados relacional amplamente adotado, a estrutura do banco de dados foi implementada conforme as necessidades do projeto. A seguir, apresentamos a estrutura das tabelas principais conforme a imagem fornecida:

- **Tabela Usuário:** Armazena informações dos usuários, com campos como *id\_usuario* (integer), *nome* (char), *email* (char), e *senha* (char).
- **Tabela Análise:** Registra cada análise realizada, associada a um usuário através do campo *id\_usuario* (integer) e contém um campo *data* (timestamp) para a data e hora da análise.
- **Tabela Dados:** Contém os dados coletados nas análises, com campos como *id\_dado* (integer), *id\_analise* (integer), *temperatura* (float), *frequência cardíaca* (integer), *gsr* (float), e *horário* (timestamp).

A integridade referencial entre as tabelas é mantida através do uso de chaves estrangeiras, permitindo uma relação lógica entre os usuários, suas análises e os dados coletados.



**Figura 1. Modelo Entidade-Relacionamento do Banco de Dados**

## 4. Descrição estrutural do sistema

Essa seção apresenta a descrição do sistema, como seus blocos funcionais, descrição do hardware e diagrama de interface.

### 4.1. Blocos Funcionais

Para melhor entender a relação entre as diferentes partes do sistema, foi criado um diagrama de blocos funcionais que pode ser visualizado na figura 2.

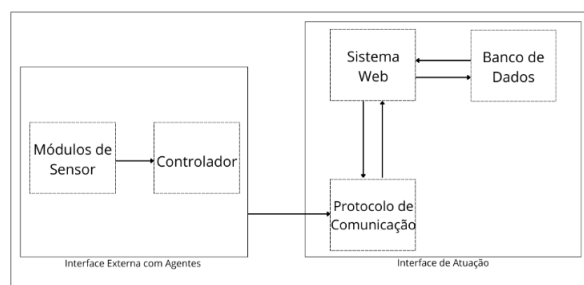
### 4.2. Especificação Estrutural

Essa subseção se refere as especificações estruturais do projeto, como firmware, comunicação entre dispositivos, interface para o usuário e resumo do algoritmo.

#### 4.2.1. Firmware

As seguintes bibliotecas foram usadas:

- Arduino.h



**Figura 2. Diagrama dos blocos funcionais.**

- WiFi.h versão 1.2.7, para comunicação WiFi
- PubSubClient.h versão 2.8.0
- OneWire.h versão 2.3.7, biblioteca do sensor de temperatura
- DallasTemperature.h versão 3.9.0, biblioteca do sensor de temperatura
- PulseSensorPlayground.h versão 1.9.1, biblioteca do sensor de frequência cardíaca

#### 4.2.2. Comunicação entre Dispositivos

A comunicação eficiente entre dispositivos é um aspecto crucial em qualquer sistema IoT. No projeto em questão, essa comunicação foi implementada utilizando o protocolo MQTT (Message Queuing Telemetry Transport), um protocolo de mensagens leve e baseado em publicação/assinatura, ideal para cenários com largura de banda limitada e requisitos de baixa latência.

Para gerenciar essa comunicação, escolheu-se o Mosquitto, um broker MQTT popular e de código aberto. O Mosquitto se destaca por sua leveza, facilidade de configuração e alta performance, sendo capaz de suportar milhares de conexões simultâneas. A configuração do Mosquitto no projeto permitiu a criação de um ponto central de comunicação, onde mensagens poderiam ser publicadas e consumidas de forma eficiente.

No âmbito do projeto, foram criados dois tópicos principais no broker MQTT:

1. **Tópico de Comandos:** Este tópico foi utilizado para enviar comandos do servidor para os dispositivos. Os dispositivos ESP estão inscritos neste tópico, aguardando comandos para iniciar a análise. Esse esquema permite uma interação direta e eficiente entre a aplicação web e os dispositivos, garantindo que os comandos sejam recebidos e processados em tempo real.
2. **Tópico de Dados:** O segundo tópico foi dedicado à transmissão de dados dos sensores para o servidor. Neste caso, o backend Django inscreve-se no tópico para receber os dados coletados pelos sensores. Essa abordagem garante uma coleta de dados contínua e confiável, permitindo que o sistema processe e analise as informações recebidas quase instantaneamente.

### 4.2.3. Interface para o usuário

Para a interface do usuário, foi utilizado um template da Creative Tim, que oferece soluções em Bootstrap. Este template foi adaptado para atender às necessidades específicas do nosso projeto Figura 3. A escolha dessa ferramenta visou proporcionar uma experiência de usuário refinada e responsiva, garantindo uma interface atraente e funcional.

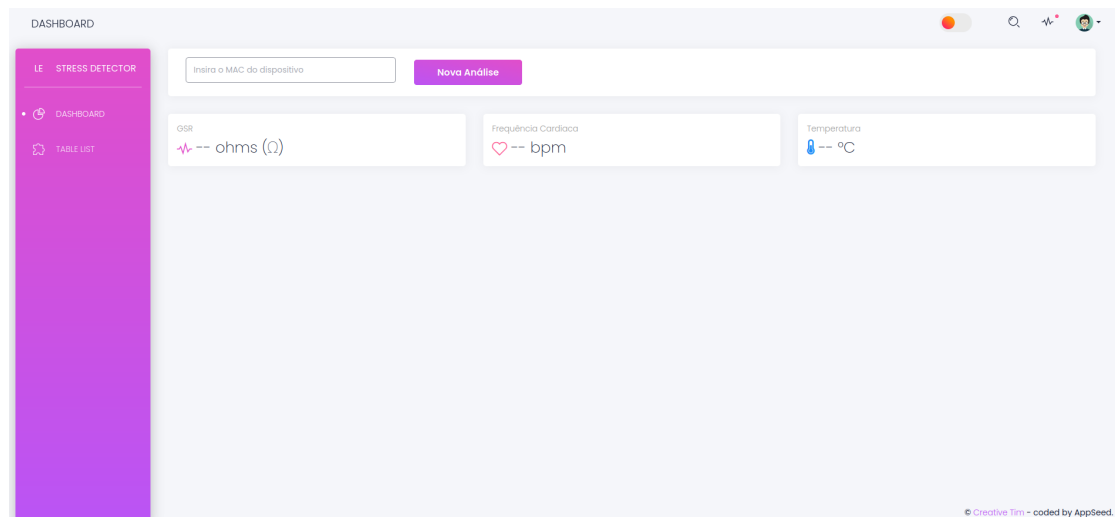


Figura 3. Interface principal do usuário.

### 4.2.4. Especificação do algoritmo

O protótipo conta com os seguintes programas: coleta de dados dos três sensores e serviço de requisição a um banco de dado. O procedimento de coleta de dados consiste na aquisição da temperatura corporal, resistência elétrica da pele e frequência cardíaca do usuário. O serviço de requisição de Banco de Dados baseia-se em fornecer ao usuário, o acesso aos seus próprios dados juntamente com a aplicação web, onde os dados são visualizados em tempo real durante uma análise.

## 4.3. Descrição do hardware

Nessa seção, cada componente do hardware será descrita, com especificações de tensão e suas pinagens.

### 4.3.1. ESP-WROOM-32

O ESP32 é uma placa de desenvolvimento com WiFi e Bluetooth, a programação pode ser desenvolvida no Arduino IDE, essa conectividade e facilidade faz o dispositivo ser muito utilizado em projetos de IoT.

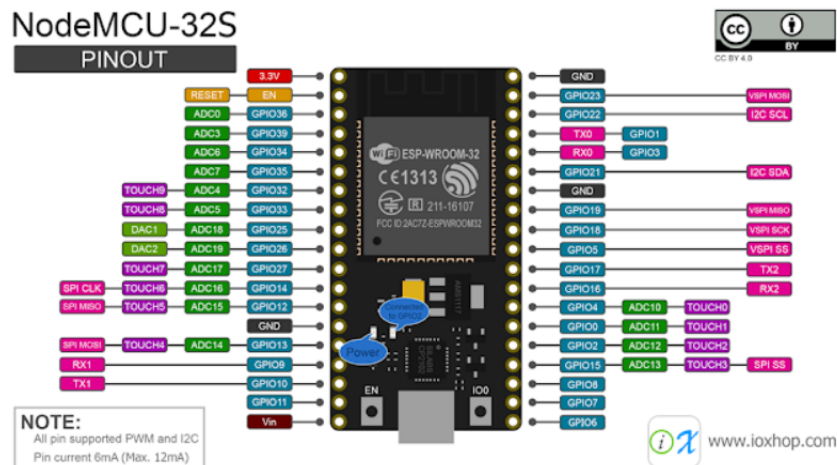


Figura 4. Figura esquemática do ESP32. [K]

ESP32	
CPU	Xtensa® Dual-Core 32-bit LX6
ROM	448 KBytes
RAM	520 Kbytes
Flash	4 MB
Clock máximo	240MHz
Wireless padrão	802.11 b/g/n
Conexão Wifi	2.4Ghz (máximo de 150 Mbps)
Bluetooth	BLE 4.2

#### 4.3.2. DS18B20

O sensor de temperatura DS18B20, que será usado do tipo sonda, é um sensor que permite a medição de temperatura na faixa de -55°C a 125°C e com precisão de  $\pm 0,5^\circ\text{C}$  na faixa de medição de -10°C a 85°C e funciona com tensão de operação 3-5,5V.



Figura 5. Figura do DS18B20

DS18B20	
Fio	Função
Vermelho	VCC
Preto	Ground
Amarelo	Data



### 4.3.3. Frequência Cardíaca

O sensor de frequência cardíaca possui um LED e um circuito de remoção de ruído, atrás do LED tem um sensor fotodetector. A medição do batimento cardíaco é feita considerando que antes do batimento há um aumento do volume nos fios capilares, e após, há uma diminuição, esse fenômeno pode ser captado numa técnica chamada fotopletismo-grafia, comumente abreviada para PPG, em que o LED ilumina a pele e o fotodetector capta quanta luz é refletida. O sensor HW-827 funciona com tensão de operação entre 3-5.5V e produz uma saída do tipo analógica.

HW-827	
Pino	Função
1	Ground
2	VCC
3	Pino analógico



**Figura 6. Figura do sensor de frequência**

### 4.3.4. GSR

O sensor GSR ou sensor de resposta galvânica da pele é responsável por medir a resistência elétrica da pele por meio de dois eletrodos que devem ser colocados em dois dedos diferentes da mesma mão. Sua tensão de operação varia entre 3.3-5V e seu sinal de entrada é a resistência do suor na pele e seu sinal de saída é do tipo analógica e é proporcional a tensão.

GSR	
Pino	Função
1	Ground
2	VCC
3	Data Analógico



**Figura 7. Figura do GSR**

## 4.4. Diagrama de interface

O seguinte diagrama de interface foi montado no software Proteus, nele é possível ver os componentes e sua ligação. Todos os 3 sensores possuem um pino/fio para alimentação, que será conectada ao pino que fornece 3.3V do ESP32, um pino/fio para ground também ligados ao mesmo pino de GND do ESP32 e um pino/fio para dados que será conectada a uma porta IO.

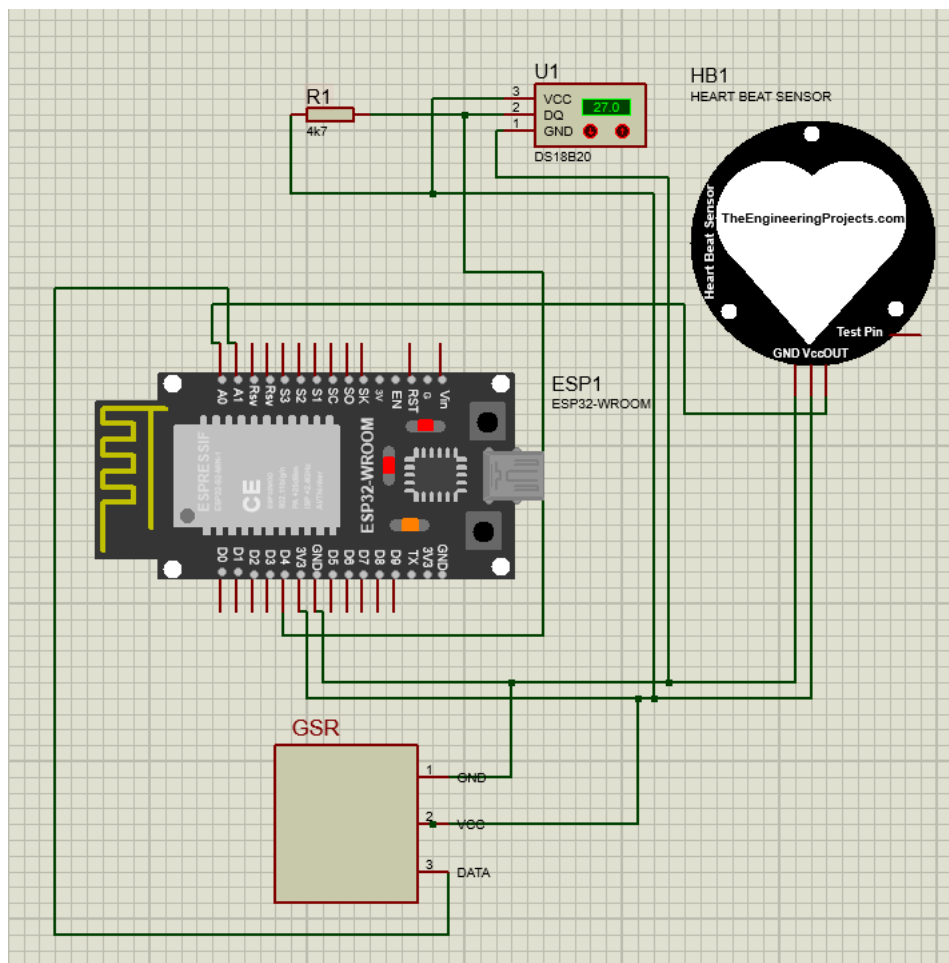


Figura 8. Diagrama de interface.

## 5. Prototipação

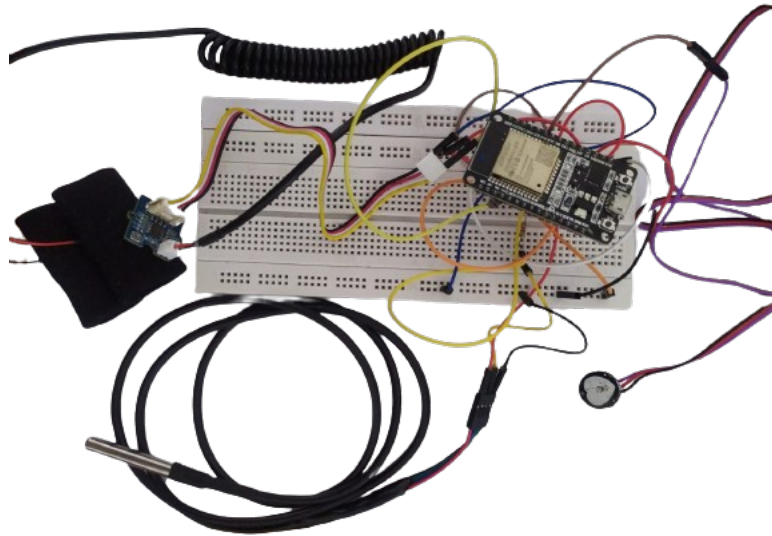
Na figura 9 é possível observar o protótipo desenvolvido.

Primeiramente, a conexão Wifi foi estabelecida pelo código. Então, o usuário precisa entrar na aplicação web, se logar, então fornecer o código MAC do dispositivo. Depois disso, apertar o botão "iniciar coleta", os dados serão coletados por 1 minuto e a última coleta é salva e mostrada ao usuário na aplicação web.

## 6. Conclusão

O protótipo se mostrou eficiente na captura de dados dos três sensores e conexão com o servidor. Sendo o resultado da implementação eficiente de conceitos da matéria de Projetos de Sistemas Ubíquos.

Para trabalhos futuros, recomenda-se a miniaturização para a criação de um vestível, a inclusão de mais sensores para a maior precisão do protótipo ou a implementação de um código que apresente na aplicação web uma média das leituras em 1 minuto, excluindo os valores extremos, para um resultado mais preciso.



**Figura 9. Protótipo.**

## Referências

- Arora, J. and Meumeu Yomsi, P. (2019). Wearable sensors based remote patient monitoring using iot and data analytics. *U.Porto Journal of Engineering*, 5:34–45.
- Benfante, A., Di Tella, M., Romeo, A., and Castelli, L. (2020). Traumatic stress in healthcare workers during covid-19 pandemic: A review of the immediate impact. *Frontiers in Psychology*, 11:569935.
- Chalabianloo, N., Can, Y. S., Umair, M., Sas, C., and Ersoy, C. (2022). Application level performance evaluation of wearable devices for stress classification with explainable ai. *Pervasive and Mobile Computing*, 87:101703.
- Cholis, B. N., Febian, S., and Ediana, S. (2021). Measurement device for stress level and vital sign based on sensor fusion. *Healthc Inform Res*, 27(1):11–18.
- Gogate, U. and Bakal, J. (2019). Hunger and stress monitoring system using galvanic skin response. *Indonesian Journal of Electrical Engineering and Computer Science*, 13:861–865.
- K, F. Esp32: Detalhes internos e pinagem.
- Osório, F. L., Silveira, I. L., Pereira-Lima, K., Crippa, J. A., Hallak, J. E., Zuardi, A. W., and Loureiro, S. R. (2021). Risk and protective factors for the mental health of brazilian healthcare workers in the frontline of covid-19 pandemic. *Frontiers in Psychiatry*, 12.
- Qadri, Y. A., Nauman, A., Zikria, Y. B., Vasilakos, A. V., and Kim, S. W. (2020). The future of healthcare internet of things: A survey of emerging technologies. *IEEE Communications Surveys Tutorials*, 22(2):1121–1167.
- Singh, R., Gehlot, A., Rashid, M., Saxena, R., Akram, S. V., Alshamrani, S. S., and AlGhamdi, A. S. (2021). Cloud server and internet of things assisted system for stress monitoring. *Electronics*, 10(24).

- Singh, R., Gehlot, A., Saxena, R., Alsubhi, K., Anand, D., Noya, I. D., Akram, S. V., and Choudhury, S. (2023). Stress detector supported galvanic skin response system with iot and labview gui. *Computers, Materials & Continua*, 74(1):1217–1233.
- Villarejo, M. V., Zapirain, B. G., and Zorrilla, A. M. (2012). A stress sensor based on galvanic skin response (gsr) controlled by zigbee. *Sensors*, 12(5):6075–6101.