

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

LUCAS FELIPE KAVADA TANAKA

GABRIEL VIEIRA GANZERT

**CASA SEGURA: MONITORAMENTO DE GRANDEZAS VIA APLICATIVO
CELULAR**

CURITIBA

2023

**LUCAS FELIPE KAVADA TANAKA
GABRIEL VIEIRA GANZERT**

**CASA SEGURA: MONITORAMENTO DE GRANDEZAS VIA APLICATIVO
CELULAR**

Safe house: Monitoring quantities thru mobile application

Trabalho de Conclusão de Curso de Graduação apresentado como requisito para obtenção do título de Bacharel em Engenharia Eletrônica do Curso de Bacharelado em Engenharia Eletrônica da Universidade Tecnológica Federal do Paraná.

Orientador: Prof. Dr. Hermes Irineu del Monego

Coorientador: Prof. Dr. Bruno Sens Chang

CURITIBA

2023



[4.0 Internacional](https://creativecommons.org/licenses/by/4.0/)

Esta licença permite compartilhamento, remixe, adaptação e criação a partir do trabalho, mesmo para fins comerciais, desde que sejam atribuídos créditos ao(s) autor(es). Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.

**LUCAS FELIPE KAVADA TANAKA
GABRIEL VIEIRA GANZERT**

**CASA SEGURA: MONITORAMENTO DE GRANDEZAS VIA APLICATIVO
CELULAR**

Trabalho de Conclusão de Curso de Graduação
apresentado como requisito para obtenção do
título de Bacharel em Engenharia Eletrônica
do Curso de Bacharelado em Engenharia
Eletrônica da Universidade Tecnológica Federal
do Paraná.

Data de aprovação: 22/junho/2023

Hermes Irineu Del Monego
Doutor
Universidade Tecnológica Federal do Paraná

Bruno Sens Chang
Doutor
Universidade Tecnológica Federal do Paraná

Omero Francisco Bertol
Mestre
Universidade Tecnológica Federal do Paraná

Luiz Fernando Copetti
Mestre
Universidade Tecnológica Federal do Paraná

**CURITIBA
2023**

Dedicamos este trabalho primeiramente a Deus, a todo curso de Engenharia Eletrônica da Universidade Tecnológica Federal do Paraná e aos nossos colegas e professores que nos auxiliaram nessa jornada.

AGRADECIMENTOS

Agradecemos primeiramente a Deus por estar sempre cuidando de nós, aos nossos familiares e amigos que sempre nos apoiaram e ajudaram nos momentos de dificuldades e incertezas, aos nossos professores orientadores, Hermes Monego e Bruno Chang pela orientação e auxílio durante toda etapa de TCC e a todas as pessoas que, de alguma forma, contribuíram, de forma positiva e significativa, para a apresentação deste trabalho.

Só se pode alcançar um grande êxito quando
nos mantemos fiéis a nós mesmos (Friedrich
Nietzsche).

RESUMO

Muitas pessoas ao saírem de suas casas, seja para ir ao trabalho e voltar no mesmo dia, ou para viajar durante longos períodos, não podem contar com outra pessoa para permanecer na residência e garantir que tudo está em ordem. O mesmo se aplica para pessoas que possuem casa em outras cidades, mas não têm disponibilidade para ir pessoalmente até elas com frequência. Sendo assim, o presente trabalho tem como objetivo desenvolver um sistema de monitoramento de grandezas de possíveis riscos à residência ou ao usuário em si, auxiliando este na gestão de sua residência com visualização através de aplicativo celular, detectando possíveis ameaças, sendo elas: vazamento de água, vazamento de gases inflamáveis ou presença de fumaça, e temperatura acima do normal. Para isso, será utilizado um sensor de fluxo de água que deverá ser instalado na caixa d'água, um sensor de gás que deverá ser instalado na cozinha e um sensor de temperatura. O sensor de fluxo de água é acoplado junto ao microcontrolador central, que possui integrado um módulo 2G, sendo este responsável pelo envio dos dados coletados e processados para o servidor. O sensor de gás e o sensor de temperatura são acoplados a um microcontrolador periférico menos custoso que não possui módulo 2G, porém que se comunica com o microcontrolador central através de BLE, de forma que este possa ler e enviar para o servidor os dados do periférico. O servidor então escreve em um banco de dados que se encontra na nuvem, de forma que o aplicativo celular possa ler todas as informações geradas pelos microcontroladores na residência do usuário. Essas informações são utilizadas para elaboração de uma aba no aplicativo de forma que o usuário possa verificar o estado de todos alertas em sua residência, isto é, se eles ocorreram ou não, bem como o horário de ocorrência. Além disso, gráficos das medições por horas ou dias são disponibilizados ao usuário por meio de outra aba no aplicativo. O projeto conta, também, com hardwares para detecção de queda de energia que acionará, de forma automática, baterias externas para que o sistema continue funcionando.

Palavras-chave: internet das coisas; monitoramento; detecção de ameaças; gestão da residência.

ABSTRACT

When many people leave their homes, either to go to work and return the same day, or to travel for long periods, they cannot count on someone else to stay at home and ensure that everything is in order. The same applies to people who own a home in other cities, but are not available to visit them in person frequently. Thus, the objective of this work is to develop a monitoring system for the magnitude of possible risks to the residence or to the user itself, helping him in the management of his residence with visualization through a mobile application, detecting possible threats, namely: water leakage, leak of flammable gases or the presence of smoke, and temperature above normal. For this, a water flow sensor will be used to be installed in the water tank, a gas sensor to be installed in the kitchen and a temperature sensor. The water flow sensor is coupled to the central microcontroller, which has an integrated 2G module, which is responsible for sending the collected and processed data to the server. The gas sensor and the temperature sensor are coupled to a less expensive peripheral microcontroller that does not have a 2G module, but which communicates with the central microcontroller through BLE, so that it can read and send peripheral data to the server. The server then writes to a database located in the cloud, so that the mobile application can read all the information generated by the microcontrollers at the user's home. This information is used to create a tab in the application so that the user can check the status of all alerts in his home, that is, whether they occurred or not, as well as the time of occurrence. In addition, charts of measurements by hours or days are available to the user through another tab in the application. The project also has hardware to detect power outages that will automatically activate external batteries so that the system continues to function.

Keywords: internet of things; monitoring; threat detection; residence management.

LISTA DE FIGURAS

Figura 1 – Exemplo de produto com sensor de gás	16
Figura 2 – Efeito Hall	21
Figura 3 – Exemplo de divisão de uma aplicação em front-end e back-end	22
Figura 4 – Exemplo do funcionamento de um ORM	24
Figura 5 – O modelo básico de protocolo de transferência de correio simples (SMTP)	26
Figura 6 – ESP32	28
Figura 7 – ESP32 TTGO T-Call	29
Figura 8 – Sensor MQ-2	29
Figura 9 – Elemento sensor do MQ-2	30
Figura 10 – Sensor YF-S201	31
Figura 11 – Sensor LM35	31
Figura 12 – Relé para circuito de alimentação	33
Figura 13 – Circuito para adequação do sensor MQ-2 - a	34
Figura 14 – Circuito para adequação do sensor MQ-2 - b	34
Figura 15 – Tabela do serviço BLE do microcontrolador periférico	36
Figura 16 – Curvas de resposta do sensor MQ-2	37
Figura 17 – Arquitetura do sistema em camadas IOT	38
Figura 18 – Modelo relacional do Banco de Dados	39
Figura 19 – Tela Últimas Medições dentro do aplicativo mobile	41
Figura 20 – Tela Medições do mês dentro do aplicativo mobile	42
Figura 21 – Tela Configurações dentro do aplicativo mobile	42
Figura 22 – Tela Alertas dentro do aplicativo mobile	43
Figura 23 – Jarra utilizada para as medições de água	46
Figura 24 – Configuração utilizada para as medições de água	46
Figura 25 – Requisição GET feita na ferramenta insomnia para o servidor e seu re- torno em JSON	48
Figura 26 – Interface da ferramenta BeeKeeper	48
Figura 27 – Aplicativo aberto no simulador Nox	49

LISTA DE TABELAS

Tabela 1 – Medições com ar limpo	44
Tabela 2 – Medições com presença de butano	45
Tabela 3 – Medições de temperatura	45
Tabela 4 – Medições de volume de água	45
Tabela 5 – Custos para desenvolvimento	49

LISTA DE SIGLAS

API	Interface de Programação de Aplicação, do inglês, <i>Application Programming Interface</i>
GLP	Gás Liquefeito de Petróleo
HTTP	Protocolo de Transferência de Hipertexto, do inglês <i>HyperText Transfer Protocol</i>
ORM	Mapeamento Objeto-Relacional, do inglês <i>Object-Relational Mapping</i>
SMTP	Protocolo de Transferência de Correio Simples, do inglês <i>Simple Mail Transfer Protocol</i>
SQL	Linguagem de Consulta Estruturada, do inglês <i>Structured Query Language</i>

SUMÁRIO

1	INTRODUÇÃO	14
1.1	Considerações iniciais	15
1.2	Objetivos	15
1.2.1	Objetivo geral	15
1.2.2	Objetivos específicos	16
1.3	Justificativa	16
2	REFERENCIAL TEÓRICO	18
2.1	Grandezas a serem monitoradas	18
2.1.1	GLP	18
2.1.2	Fumaça	18
2.1.3	Temperatura	19
2.1.4	Volume	19
2.2	Internet das coisas	19
2.3	Sistema microcontrolado	19
2.4	Medições	20
2.4.1	QUIMIORESISTOR	20
2.4.2	EFEITO HALL	20
2.5	Front-end	21
2.5.1	React Native	21
2.6	Back-end	22
2.6.1	API	23
2.6.2	Node.js	23
2.6.3	O.R.M	23
2.7	Banco de dados	24
2.7.1	MySQL	25
2.8	Protocolos de Comunicação	25
2.8.1	HTTP	25
2.8.2	SMTP	26
3	DESENVOLVIMENTO DO SISTEMA	27
3.1	DEFINIÇÃO DOS MICROCONTROLADORES	27

3.1.1	ESP32	27
3.1.2	TTGO T-Call	28
3.2	DEFINIÇÃO DOS SENSORES	29
3.2.1	MQ-2	29
3.2.2	YF-S201	30
3.2.3	LM35	31
3.3	DEFINIÇÃO DO MÉTODO DE COMUNICAÇÃO	32
3.4	DESENVOLVIMENTO DO HARDWARE	32
3.4.1	CIRCUITO DE ALIMENTAÇÃO	32
3.4.2	CIRCUITO DO PERIFÉRICO	33
3.4.3	CIRCUITO DA CENTRAL	33
3.5	IMPLEMENTAÇÃO DO FIRMWARE	34
3.5.1	LÓGICA BÁSICA DO SISTEMA	35
3.5.2	AQUISIÇÃO DAS MEDIDAS	36
3.6	IMPLEMENTAÇÃO DO SERVIDOR	38
3.6.1	BANCO DE DADOS	38
3.6.2	APLICAÇÃO NODEJS	39
3.7	IMPLEMENTAÇÃO DO APLICATIVO	40
3.7.1	ULTIMAS MEDIÇÕES	40
3.7.2	MEDIÇÕES DO MÊS	41
3.7.3	CONFIGURAÇÕES	41
3.7.4	ALERTAS	43
4	MÉTODOS E DISCUSSÃO DOS RESULTADOS	44
4.1	TESTES DE HARDWARE	44
4.2	TESTES DAS MEDIÇÕES	44
4.3	TESTE DA COMUNICAÇÃO PERIFÉRICO-CENTRAL	47
4.4	TESTE DA COMUNICAÇÃO 2G DA CENTRAL	47
4.5	TESTES DO SERVIDOR	47
4.6	TESTES DO APLICATIVO	47
4.7	TESTE DA GERAÇÃO DE ALERTAS	48
4.8	ANÁLISE DE CUSTOS	49

5	CONSIDERAÇÕES FINAIS	50
5.1	IMPLEMENTAÇÕES FUTURAS	50
	REFERÊNCIAS	52
	APÊNDICE A ESQUEMÁTICO DO HARDWARE DO PERIFÉRICO . . .	57
	APÊNDICE B ESQUEMÁTICO DO CIRCUITO DE ALIMENTAÇÃO	59

1 INTRODUÇÃO

A segurança da residência é uma preocupação das mais importantes de todos aqueles que possuem uma casa própria, ou mesmo daqueles que são inquilinos em apartamentos. E cuidar desse aspecto das moradias não é uma tarefa fácil, especialmente quando não é possível estar presente nelas.

O Gás Liquefeito de Petróleo (GLP), conhecido popularmente no Brasil como gás de cozinha, está presente na maioria das residências que se utilizam de fogão a gás. Neste tipo de fogão, o gás é expelido pela boca do mesmo, e então é necessária uma pequena faísca para produzir a chama, a ser utilizada para cozinhar os alimentos. Embora bastante simples, esse tipo de fogão também apresenta perigos, uma vez que o GLP é altamente inflamável e pode ser vazado acidentalmente desde o botijão de gás até a boca do fogão, podendo ocasionar asfixias ou até incêndios e explosões em casos mais graves, causando óbitos ou queimaduras severas no corpo de pessoas (GLOBO, 2022). Entre 2021 e 2022, o Corpo de Bombeiros do Paraná registrou mais de 300 ocorrências de vazamento de gás de cozinha, inclusive com registro de uma vítima fatal (AERP, 2022). No Ceará, por exemplo, esse número é muito mais preocupante, sendo registradas 1650 ocorrências somente até Outubro de 2022 (CBMCE, 2022).

Outro problema bastante inconveniente em residências pode ser o vazamento de água. Além da possibilidade de causar inundações se não detectada rapidamente, essa ocorrência pode ter um impacto bastante indesejável financeiramente. Em maio de 2022, a Sanepar aplicou um reajuste de 4,96% na tarifa de água e esgoto. Com a mudança, a tarifa mínima deve passar para R\$ 81,45 (LONDRINA, 2022). Esse aumento gradativo tem se tornado, cada vez mais, um empecilho na vida de muitos brasileiros. Além do ponto de vista econômico, deve ser analisado também os desperdícios. O conceito de sustentabilidade tem se tornado cada vez mais relevante nos dias atuais, em que uma das principais causas é o aumento da escassez da água (DW, 2022).

Esses problemas podem ser detectados quando alguma pessoa está presente na residência, seja através do olfato na presença de vazamento de GLP ou pela audição ou mesmo visão no vazamento de água, e ainda assim podem ser difíceis de serem percebidos, ou percebidos tardiamente. Além disso, quando a moradia fica sem nenhuma pessoa presente para cuidar da mesma, a detecção desses problemas por esses métodos se torna inviável, a não ser quando essas ocorrências já causaram grandes impactos, como incêndios, explosões ou inundações.

Sendo assim, o presente trabalho de conclusão de curso tem como objetivo o projeto de um sistema microcontrolado de monitoramento de grandezas possivelmente prejudiciais à integridade da residência do usuário, ou até do usuário em si, para que este possa visualizar o estado de sua casa em qualquer momento através de um aplicativo celular.

1.1 Considerações iniciais

Buscando um meio de identificar as ameaças descritas anteriormente, este trabalho visa o projeto e desenvolvimento de um sistema resiliente à queda de energia, capaz de monitorar grandezas que possam apresentar eventuais riscos à uma residência do usuário, de forma que este, ao verificar as ocorrências através do aplicativo, possa tomar as devidas providências o mais rápido possível. Para isso, o sistema é dividido em três partes:

- o sistema microcontrolado, fixo na moradia do usuário;
- o back-end, rodando na nuvem;
- o aplicativo com o qual o usuário interage.

O sistema microcontrolado é dividido em dois módulos, separados fisicamente: o periférico, e o central. O módulo periférico, instalado preferencialmente na cozinha, tem o objetivo de detectar a presença de gases estranhos ao ambiente residencial, como o propano no caso de um vazamento de GLP, ou do monóxido de carbono na presença de um incêndio, além de obter a temperatura do local. Como método de comunicação com a central, foi escolhido o BLE (*Bluetooth Low Energy*), de forma que ao obter as medidas e verificar a ocorrência de alertas, o periférico então escreve nas respectivas características, para que então a central possa ler destas. A central, por sua vez, é responsável pelo envio para o servidor das informações obtidas pelo periférico, além do monitoramento do vazamento de água, sendo acoplada a um sensor de fluxo de água instalado em um cano de saída da caixa d'água. Buscando a resiliência do sistema, optou-se pelo uso da tecnologia 2G para transmissão dos dados, de forma a não depender da presença de Wi-Fi. Além disso, ambos módulos contam com um hardware para detecção de queda de energia, chaveando para o uso de uma bateria externa caso seja necessário, mantendo o sistema totalmente funcional.

O back-end é responsável por obter as informações do sistema microcontrolado na residência, e armazenar as informações em um banco de dados rodando na nuvem, de forma que o aplicativo possa interagir com esse banco e exibir as informações para o usuário.

1.2 Objetivos

1.2.1 Objetivo geral

Desenvolver um sistema inteligente de detecção de ameaças dentro de uma residência com intuito de auxiliar os usuários na gestão da mesma.

1.2.2 Objetivos específicos

- Desenvolver um módulo microcontrolado periférico para processamento e envio dos dados coletados para a central.
- Desenvolver um módulo microcontrolado central para coletar os dados enviados pelo periférico e escrever no banco de dados na nuvem por meio de APIs.
- Desenvolver um aplicativo mobile para visualização dos dados coletados e configuração dos alertas.

1.3 Justificativa

A preocupação com a segurança da residência dos cidadãos é um tópico bastante explorado – e até saturado – pelo mercado. Existem diversos produtos que realizam o monitoramento de possíveis invasores através de detectores de movimento ou sensores de presença. Também há produtos que realizam o monitoramento de vazamentos de gás, ou também de água, de forma que o usuário possa visualizar as ocorrências pelo aplicativo, ou simplesmente por meio de avisos sonoros ou visuais. Entretanto, não foi encontrado nas pesquisas algum produto que reúna em um só sistema a verificação desses dois problemas. Além disso, todos os produtos encontrados que se utilizam de uma interface gráfica para interação com o usuário necessitam de uma rede Wi-Fi ativa para a transmissão dos dados e, também por consequência, da rede elétrica fornecendo energia normalmente para alimentação do sistema. Como exemplo disso, tem-se o sensor HISSGL da empresa Geonav, conforme Figura 1.

Figura 1 – Exemplo de produto com sensor de gás



Fonte: (AMAZON, 2022).

Outro produto semelhante é o Detector de Gás Smart Intelbras IDG 620, que emite um alarme sonoro e uma notificação no aplicativo Izy quando da detecção de um vazamento. Entretanto, para receber os alertas via aplicativo é necessário comprar separadamente um hub

de automação, que é então sincronizado com o detector de gás, o que aumenta bastante o custo total para uma solução de monitoramento via internet.

É evidente que esses produtos estão sujeitos à quedas de conexão da rede Wi-Fi, além de eventuais quedas de energia da rede elétrica. Sendo assim, o presente trabalho tem como objetivo um sistema que, além de verificar ocorrências de vazamento de gás ou presença de fumaça e vazamento de água, seja resiliente à quedas de energia e que portanto não necessite de uma rede Wi-Fi para funcionar normalmente. Tais aspectos aumentam o custo final do sistema, uma vez que para atingir esse objetivo é necessário um módulo GSM juntamente à um chip ativo, e uma bateria externa para cada módulo microcontrolado, sendo esse um tradeoff entre custo e confiabilidade do sistema. Tal troca foi considerada benéfica no presente trabalho, uma vez que se trata de um sistema que visa garantir a segurança e integridade da moradia do usuário, ou até mesmo das pessoas presentes na casa ou vizinhos, em casos mais graves como incêndios, além do fato de que, embora quedas de energia da rede elétrica sejam mais raras, quedas de conexão Wi-Fi ocorrem com alguma frequência na residência de diversas pessoas.

Além disso, este trabalho apresenta uma grande oportunidade para aplicar diferentes conhecimentos adquiridos durante o curso de engenharia eletrônica, integrando principalmente as áreas de desenvolvimento de firmware, de software e de hardware.

2 REFERENCIAL TEÓRICO

Neste capítulo serão abordados os principais conceitos que norteiam este projeto. Os assuntos abordados serão sobre as grandezas a serem monitoradas e sobre os microcontroladores, sensores, banco de dados, protocolo de comunicação e back-end utilizados neste trabalho.

2.1 Grandezas a serem monitoradas

2.1.1 GLP

O Gás Liquefeito de Petróleo (GLP), conhecido principalmente como “gás de cozinha” ou também como “gás de botijão”, é uma mistura de gases de hidrocarbonetos, formando um combustível destinado principalmente para uso doméstico e industrial.

É composto majoritariamente por uma mistura de hidrocarbonetos contendo de três a quatro átomos de carbono, que embora gasoso nas Condições Normais de Temperatura e Pressão (CNTP), pode ser liquefeito sobre compressão ou resfriamento, o que otimiza o transporte e armazenamento. Essa mistura é feita principalmente de propano (hidrocarboneto com três átomos de carbono) e butano (hidrocarboneto com quatro átomos de carbono). Esses gases, além de altamente inflamáveis, são também inodoros, de forma que o cheiro que é possível sentir quando da liberação do GLP vem de uma substância a base de enxofre adicionada à ele. Além disso, embora não seja venenoso, o GLP é asfíxiante, pois é mais denso que o ar, de forma que este é empurrado gradualmente para fora do ambiente (FOGAÇA, 2022).

2.1.2 Fumaça

Denomina-se fumaça a suspensão de partículas na atmosfera resultantes de uma combustão que não foram queimadas durante o processo. Constitui um produto indesejado da combustão, como em queimadas e em motores de gasolina por exemplo, de forma que quando essa ocorre de maneira completa e correta, os únicos resíduos resultantes são água e dióxido de carbono, além de outros compostos.

A simples inalação da fumaça é uma das principais causas de óbitos em casos de incêndio, devido a presença de componentes tóxicos como o monóxido de carbono, além de pequenas partículas sólidas que causam o bloqueio de vias respiratórias, e portanto a asfixia de quem as inalar (NÓBREGA, 2022).

2.1.3 Temperatura

É uma grandeza física que mede o grau de agitação das moléculas, ou simplesmente a energia cinética dessas, que constituem um corpo ou sistema em equilíbrio térmico, de forma que quanto maior esse grau de agitação, mais quente o corpo estará e portanto maior é a temperatura, e vice-versa (SILAS, 2022).

2.1.4 Volume

O volume de um corpo é simplesmente a quantidade de espaço que este corpo ocupa. É medido em unidades de tamanhos cúbicos, como mm^3 , cm^3 , m^3 , entre outros. Outra unidade bastante utilizada é o Litro, de maneira que 1 m^3 (um metro cúbico) corresponde a 1000 Litros (GOUVEIA, 2022).

2.2 Internet das coisas

O termo IoT, ou Internet das Coisas, refere-se à rede coletiva de dispositivos conectados e à tecnologia que facilita a comunicação entre os dispositivos e a nuvem, bem como entre os próprios dispositivos. Graças ao advento de chips de computador baratos e telecomunicações de alta largura de banda, agora temos bilhões de dispositivos conectados à Internet. Isso significa que dispositivos do dia a dia, como escovas de dentes, aspiradores, carros e máquinas, podem usar sensores para coletar dados e responder de forma inteligente aos usuários (AWS, 2022a).

A Internet das Coisas integra “coisas” cotidianas à Internet. Engenheiros de computação vêm adicionando sensores e processadores a objetos do cotidiano desde a década de 1990. No entanto, o progresso foi inicialmente lento porque os chips eram grandes e volumosos. Chips de computador de baixa potência chamados etiquetas RFID foram usados pela primeira vez para rastrear equipamentos caros. À medida que os dispositivos de computação diminuíram de tamanho, esses chips também se tornaram menores, mais rápidos e mais inteligentes ao longo do tempo (AWS, 2022a).

2.3 Sistema microcontrolado

De maneira geral, toda aplicação de IoT se utiliza de um sistema microcontrolado. Os microcontroladores são constituídos de uma CPU (unidade central de processamento), memória de dados, memória de código, além de periféricos para interação com o mundo externo, como canais ADC e entradas e saídas digitais, permitindo integração com elementos como atuadores, sensores e módulos de comunicação em um único sistema.

Por serem dispositivos programáveis, seu uso não se limita a aplicações específicas, podendo ser empregado em praticamente qualquer circuito que demande algum grau de processamento de informações. Seu uso simplifica os projetos eletrônicos, pois reduz consideravelmente a quantidade de componentes necessários, pois toda a lógica é programada. Atualmente os microcontroladores estão integrando diversos periféricos com o objetivo de centralizar a maior parte das funções dos circuitos, tornando-os ainda muito mais versáteis (ROCHA, 2012).

2.4 Medições

2.4.1 QUIMIORESISTOR

Um quimioresistor é um material cuja resistência elétrica é sensível, ou seja, variável em resposta à mudanças no ambiente químico próximo. Os quimioresistores são uma classe de sensores químicos que dependem da interação química direta entre o material sensor e o analito. O material sensor e o analito podem interagir por ligação covalente, ligação de hidrogênio ou reconhecimento molecular. Vários materiais diferentes têm propriedades quimioresistores: semicondutores de óxido de metal, alguns polímeros condutores, e nanomateriais como grafeno, nanotubos de carbono e nanopartículas (BANICA, 2012).

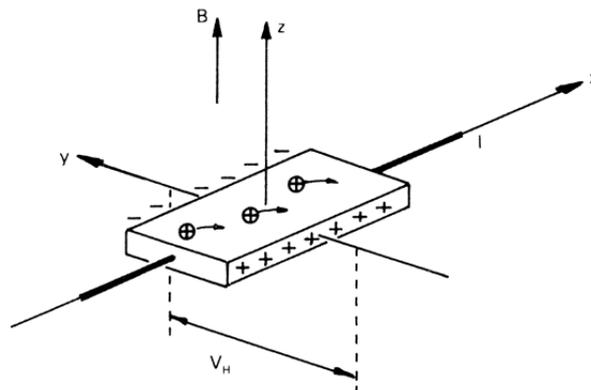
Um quimioresistor básico consiste em um material sensor que faz a ponte entre dois eletrodos, de maneira que a resistência entre os eletrodos possa ser facilmente medida. Uma vez que o material sensor tem uma resistência inerente, esta pode ser modulada pela presença ou ausência do analito. Durante a exposição, os analitos interagem com o material sensor, causando alterações na leitura da resistência. Em alguns quimioresistores, a sua resistência elétrica simplesmente indica a presença ou não do analito, já em outros a resistência é proporcional à quantidade de analito presente, o que permite que esta quantidade seja medida. (KHANNA, 2012).

2.4.2 EFEITO HALL

O efeito Hall é um fenômeno que ocorre ao aplicar uma corrente elétrica a uma placa fina feita de um material condutor, e perpendicular a esta corrente um campo magnético, resultando em uma diferença de potencial perpendicular tanto a corrente quanto ao campo magnético. Esse efeito foi descoberto em 1879 por Edwin Herbert Hall, sendo batizado com seu nome (ANTUNES, 2012).

Uma corrente I flui na direção positiva de x , e um campo magnético é aplicado na direção positiva de z , defletindo os portadores de carga positiva na direção negativa de y e os portadores de carga negativa na direção positiva de y , conforme demonstra a Figura 2. Este fenômeno resulta em potencial de Hall e em uma resistência de Hall.

Figura 2 – Efeito Hall



Fonte: BERTULANI (2005).

Através deste princípio foram criados os sensores de efeito Hall. Estes sensores consistem de um transdutor que, quando sujeito a uma variação de campo magnético, produz em sua saída uma variação de tensão. Constituem a base de diversos sensores, como por exemplo sensores de deslocamento e indicadores de nível de combustível.

2.5 Front-end

Front-end, também conhecido como "lado do cliente", é a parte responsável pela estrutura, design, conteúdo, comportamento, desempenho e capacidade de resposta de um site ou aplicação, ou seja, cuida das partes visuais e da interação entre usuários e tela. Exemplos de aplicações front-end: sites, aplicativos mobile.

2.5.1 React Native

O react native, lançado em 2015 pelo Facebook como um projeto de código aberto, é uma estrutura de aplicativo móvel popular, baseada na linguagem JavaScript, cujo objetivo é a criação de aplicativos mobiles renderizados nativamente para iOS e Android (ALURA, 2022b).

As principais vantagens de se desenvolver uma aplicação utilizando essa ferramenta são:

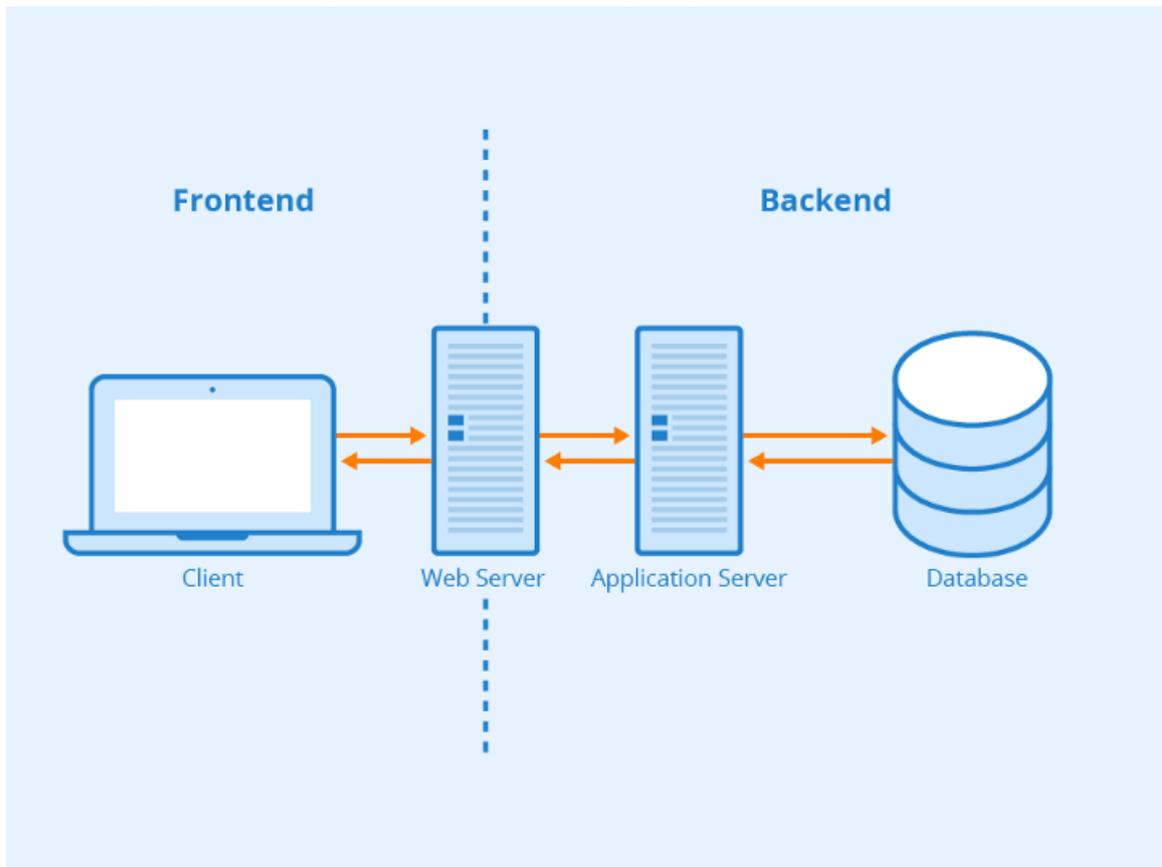
- Experiência do usuário fluída;
- Carregamento, em geral, mais rápidos;
- Traz diversas facilidades ao desenvolvedor, como alguns componentes já prontos para uso.
- Melhor performance;
- Mais seguro.

Antes do surgimento desse framework, o desenvolvimento de aplicativos mobile era algo muito complexo, pois era necessário desenvolver uma aplicação para cada sistema operacional (iOS e Android). Porém, utilizando o react native, o código pode ser reaproveitado em até 100% entre as plataformas; reduzindo, dessa forma, os custos e a duração do desenvolvimento da aplicação (BECKER, 2021).

2.6 Back-end

Back-end, também conhecido como "lado do servidor", é a parte da programação voltada aos bancos de dados e servidores. Ou seja é responsável pelas atualizações, mudanças, segurança, gerenciamento de conteúdos, estrutura, entre outras coisas da sua aplicação. É nele que os dados serão tratados, inseridos e lidos do banco de dados. A Figura 3 demonstra um exemplo da divisão de uma aplicação.

Figura 3 – Exemplo de divisão de uma aplicação em front-end e back-end



Fonte: Tekade (2021).

2.6.1 API

Interface de Programação de Aplicação, do inglês, *Application Programming Interface* (API), é um mecanismo que permite a comunicação entre dois componentes de software por meio de um conjunto de definições e protocolos. O principal objetivo dela é automatizar processos manuais e/ou permitir a criação de novas funcionalidades.

A arquitetura da API é explicada em termos de cliente e servidor. Ou seja, a aplicação que envia a solicitação é chamada de cliente e a que envia resposta é o servidor (AWS, 2022b).

2.6.2 Node.js

O node.js é um ambiente de execução javascript que atua no lado do servidor. Nele é possível a execução de aplicações desenvolvidas pela linguagem de forma autônoma sem a necessidade de utilizar um navegador (TECNOBLOG, 2020).

As principais vantagens em se utilizar esse ambiente está na sua alta escalabilidade, ou seja, o fato de ter uma grande potencialidade para suportar um número grande de conexões simultâneas. Ser multiplataforma e multi-paradigma, ou seja, permite criar aplicações para plataformas diferentes e com diferentes paradigmas (como orientado a objetos, funcional, imperativo e dirigido à eventos). E pelo fato de ser open source, tendo acesso ao seu código fonte e; desta forma, conseguir customizar suas aplicações (ALURA, 2022a).

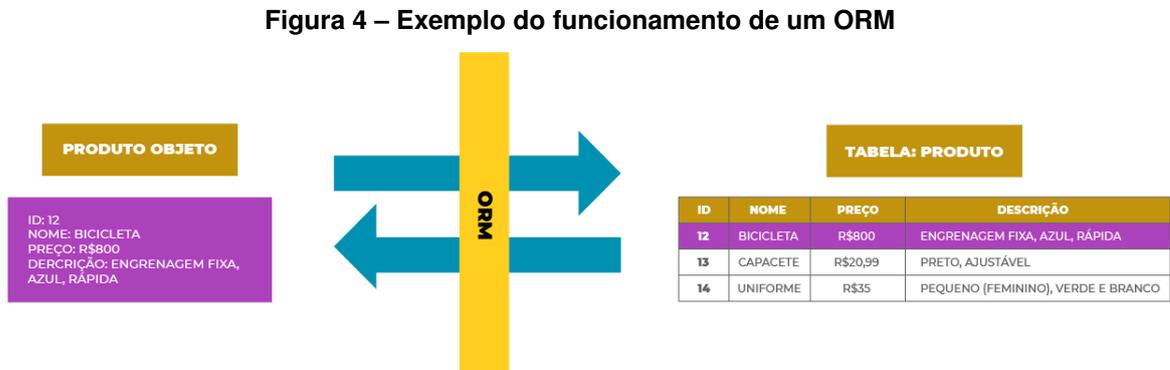
2.6.3 O.R.M

Mapeamento Objeto-Relacional, do inglês *Object-Relational Mapping* (ORM), consiste em uma técnica, cujo objetivo é aproximar o paradigma de desenvolvimento de aplicações orientadas a objetos ao paradigma do banco de dados relacional (DEVMEDIA, 2011). Ou seja, enquanto o banco de dados relacional trabalha com tabelas e relações a programação orientada a objetos trabalha com elementos como classes, propriedades, visibilidade, herança e interfaces. As principais dificuldades criadas por essas diferenças de paradigmas:

- Representação de dados e modelos, visto que estruturas são diferentes;
- Mapeamento entre a linguagem de programação e o modelo do banco de dados;
- Modelo de banco de dados relacional de integridade relacional;

As estruturas ORM definem o modo como os dados serão mapeados entre os diferentes tipos de ambientes e como serão acessados ou gravados. Trazendo, dessa forma, uma redução no tempo de desenvolvimento da aplicação e uma maior produtividade, visto que não é necessário desenvolver toda essa parte de tratativa com o banco de dados, deixando a respon-

sabilidade ao ORM (TREINAWEB, 2019). A Figura 4 demonstra um exemplo do funcionamento de um ORM.



Fonte: TREINAWEB (2019).

2.7 Banco de dados

Banco de dados é uma coleção de informações ou dados estruturados e organizados que, na maior parte das vezes, estão armazenados eletronicamente em um sistema de computador. Existem diversos tipos, dentre os principais podem ser citados:

- Relacionais: Os itens dessa estrutura são organizados como um conjunto de tabelas com colunas e linhas. A grande vantagem desse modelo é a sua eficiência e flexibilidade na consulta dos dados.
- Orientados a objetos: Os registros são armazenados no formato de objeto, como no paradigma orientado a objetos.
- Distribuídos (BDD): Os arquivos deste modelo podem estar localizados em diferentes sites e o banco de dados pode se encontrar em um servidor diferente.
- Data warehouses: Esse modelo é focado em consultas e análises rápidas, utilizando diversas fontes de dados centralizadas.
- NoSQL: No NoSQL, ou banco de dados não relacional, os dados são armazenados de forma que não há uma relação direta entre os elementos.
- Gráficos: Nesse modelo os dados são armazenados em termos de entidades e os relacionamentos entre entidades.
- OLTP: Essa estrutura foi projetada para atender vários usuários que realizarão consultas simultaneamente, portanto devem ser rápidos e analíticos.

Dentre os modelos mencionados acima, ainda existem outras estruturas que são voltadas para o âmbito científico, financeiro ou outras que são mais específicas. Em sua maioria, os bancos de dados são controlados por um sistema de gerenciamento de banco de dados (DBMS). Cujas funções são auxiliar no controle, acesso, organização e proteção das informações de uma aplicação (ORACLE, 2022).

2.7.1 MySQL

O MySQL é um sistema gerenciador de banco de dados relacional de código aberto. O modelo se utiliza da Linguagem de Consulta Estruturada, do inglês *Structured Query Language* (SQL), que é extremamente relevante quando o assunto é manipulação de dados. Ela é muito utilizada na ferramenta de Business Intelligence, e sua principal vantagem está na velocidade em que as consultas são realizadas. Em que uma grande quantidade de dados são recuperados de forma rápida e muito eficiente (TECHTUDO, 2012).

2.8 Protocolos de Comunicação

Protocolo de comunicação, para um ambiente computacional, pode ser definido como um sistema que controla e possibilita uma conexão, comunicação ou transferência de dados entre dois sistemas computacionais. Em outras palavras são "regras" que governam a sintaxe, semântica e sincronização da comunicação. Para uma aplicação IoT é importante escolher um protocolo que tenha uma comunicação rápida e confiável, visto que há um grande tráfego de dados (KITAMURA, 2022).

Dentre os protocolos mais conhecidos na indústria de IOT estão: HTTP (*HyperText Transfer Protocol*), MQTT (*Message Queue Telemetry Transport*), CoAP (*Constrained Application Protocol*) e o AMQP (*Advanced Message Queue Protocol*). Cada um possui suas particularidades e suas implementações (NAIK, 2017).

2.8.1 HTTP

Protocolo de Transferência de Hipertexto, do inglês *HyperText Transfer Protocol* (HTTP), é o protocolo de comunicação base de qualquer troca de dados na web que permite a transferência hiper-texto de um lado a outro. A comunicação sempre se inicia pelo lado do cliente, portanto é um protocolo cliente-servidor, em que as mensagens enviadas pelo cliente, na maioria das vezes um navegador da Web, são chamadas de solicitações (requests), ou requisições, enquanto que as mensagens retornadas pelo servidor são chamadas de respostas (responses). O acesso aos recursos disponibilizados pelo servidor se dá por meio da URI (*Uniform resource identifier*), que são as rotas onde os dados estarão disponíveis (MDN, 2022).

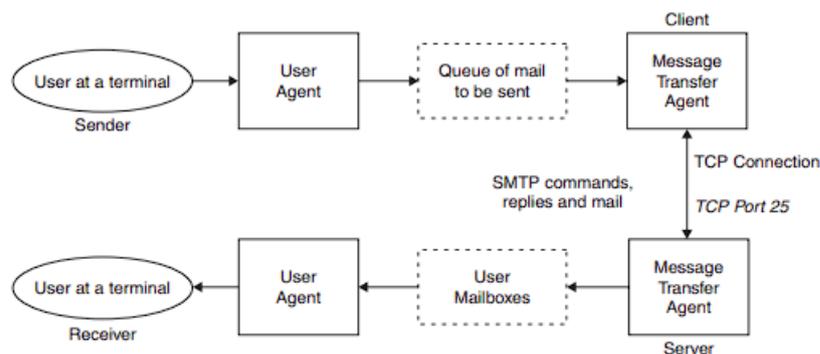
O HTTP possui alguns métodos para a padronização de sua utilização, entre eles o mais comum para se receber dados é o método “GET”, para envio de dados, é utilizado o método “POST”, para atualização completa de um dado é utilizado o “PUT”, para atualização parcial de um dado é utilizado o “PATCH” e para remoção de um dado é utilizado o “DELETE” (SHADRACK, 2022).

As informações que serão enviadas pelo cliente ao servidor, podem ser passadas pelo corpo da requisição (“body param”), para os métodos “POST” ou “PUT”, dentro de um objeto JSON. Exemplo: “nome”: “Lucas”, “idade”: 24”. Ou por parâmetro na URL (“query param”), exemplo: “http://url.com/usuario?nome=Lucas”, ou dentro da requisição da rota, exemplo: “http://url.com/usuario/Lucas”, e até mesmo pelo cabeçalho da requisição, geralmente relacionados à autorização e autenticação (ROCKETSEAT, 2020).

2.8.2 SMTP

Protocolo de Transferência de Correio Simples, do inglês *Simple Mail Transfer Protocol* (SMTP), apresentado na Figura 5, é um protocolo utilizado no envio e recebimento de e-mail. Cujo objetivo é a transferência de e-mails de forma confiável e eficiente. Ele é independente do subsistema de transmissão particular e requer apenas um canal confiável de fluxo de dados ordenados. O SMTP também é capaz de retransmitir mensagens entre ambientes de serviços de transporte (POSTEL, 1982).

Figura 5 – O modelo básico de protocolo de transferência de correio simples (SMTP)



Fonte: Riabov (2005).

3 DESENVOLVIMENTO DO SISTEMA

Visando atingir os objetivos descritos nos capítulos anteriores, as seguintes etapas foram concebidas: definição dos microcontroladores, definição dos sensores, definição do método de comunicação, desenvolvimento do hardware, desenvolvimento do firmware, teste dos sensores, implementação do servidor e desenvolvimento do aplicativo.

3.1 DEFINIÇÃO DOS MICROCONTROLADORES

Para o presente projeto, foi escolhida uma arquitetura com dois microcontroladores, denominados microcontrolador periférico e microcontrolador central. O periférico tem a responsabilidade de captar as medidas de gás e temperatura ambiente, sem necessidade de possuir comunicação por rede de telefonia celular, porém com meios de realizar uma comunicação sem fio com outro microcontrolador. O microcontrolador central, por sua vez, também deve se comunicar via tecnologia wireless com outro microcontrolador, mas também precisa ser capaz de realizar comunicação por rede de telefonia celular, de maneira a enviar as informações do sistema no qual está instalado para o servidor, além de realizar as medidas de volume de água.

3.1.1 ESP32

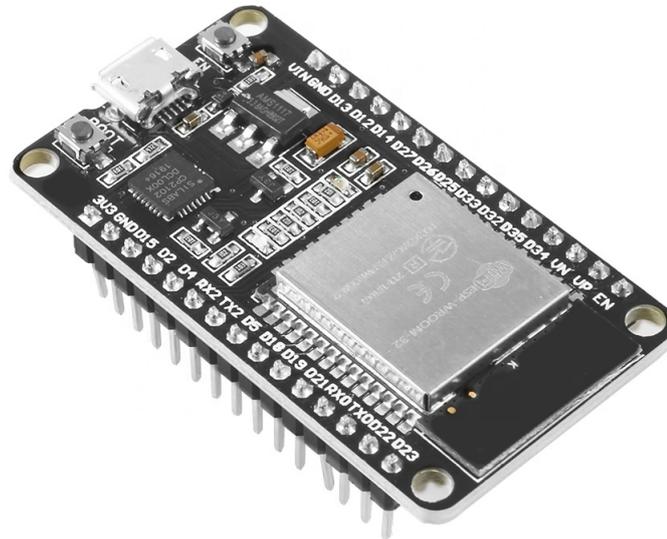
O ESP32, apresentado na Figura 6, é um microcontrolador desenvolvido pela Espressif Systems com Wi-Fi e Bluetooth integrados, enquanto possui um baixo consumo de energia, sendo ideal para aplicações de IoT. Possui versões single e dual-core 32 bits que podem operar a até 240 MHz de clock.

É uma evolução do ESP8266, também da Espressif, entregando mais memória RAM, memória FLASH, clock mais veloz, além do já mencionado módulo Bluetooth integrado, e também um custo baixo levando-se em conta todas suas funcionalidades.

Os principais recursos que este dispositivo possui são:

- Processador Xtensa 32-Bit LX6 Dual Core;
- Clock: 80 à 240 MHz;
- RTC Integrado de 8kB (Slown / Fast);
- Memória SRAM: 520kB;
- GPIO: Digital IO (36), ADC 12-Bits (16 canais), DAC 8-Bits (2 canais), Sensor Capacitivo (10 canais);
- Interfaces: Cartão SD, UART(3 canais), SPI (3 canais), SDIO, I2C (2 canais), I2S (2 canais), IR, PWM LED (2 canais) e PWM motor (3 canais);

Figura 6 – ESP32



Fonte: ALIBABABA (2022).

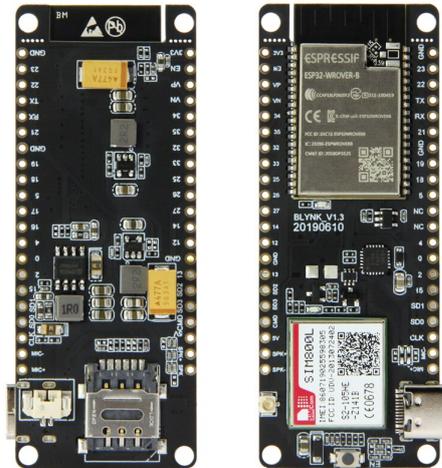
- WiFi 802.11 b/g/n: 2.4 à 2.5 GHz, com segurança WPA / WPA2 / WPA2-Enterprise / WPS;
- Bluetooth 4.2 BR / EDR e BLE (Bluetooth Low Energy).

Sendo assim, uma vez que possui ótimo poder de processamento, 16 canais ADC e BLE integrado, o ESP32 foi o microcontrolador escolhido para desempenhar a função de periférico do sistema.

3.1.2 TTGO T-Call

O TTGO T-Call, apresentado na Figura 7, é uma placa desenvolvida que combina o ESP32, possuindo todas as funcionalidades deste, junto a um módulo GSM/GPRS: o SIM800L. Dessa forma, esse dispositivo se torna ainda mais ideal para IoT, uma vez que além do Wi-Fi e Bluetooth, possui também conexão via 2G. Portanto, o TTGO se torna a escolha ideal para o desempenho da função de microcontrolador central, podendo se comunicar com outro microcontrolador via BLE, e enviar e receber pacotes via internet através do 2G, além de pinos de GPIO para receber sinais do sensor de água.

Figura 7 – ESP32 TTGO T-Call



Fonte: ALIEXPRESS (2022).

3.2 DEFINIÇÃO DOS SENSORES

3.2.1 MQ-2

O sensor MQ2 é um sensor MOS (*Metal Oxide Semiconductor*), apresentado na Figura 8, sendo um dos mais populares da série de sensores MQ. Os sensores de óxido de metal são chamados também de quimioresistores, pois a detecção ocorre de acordo com a mudança na resistência do material sensor quando exposto a gases.

Figura 8 – Sensor MQ-2

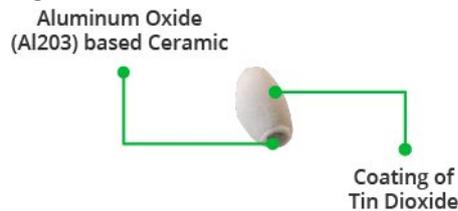


Fonte: MAMUTELETRÔNICA (2022).

O MQ2 opera em 5V e consome aproximadamente 800mW. Pode detectar concentrações de GLP, Fumaça, Álcool, Propano, Hidrogênio, Metano e Monóxido de Carbono variando de 200 a 10.000 ppm (ENGINEERS, 2022).

Este sensor pode ser separado em duas partes principais: o revestimento, na parte externa, e o elemento sensor, interno ao revestimento. O revestimento consiste de duas camadas finas de aço inoxidável, que garantem que o elemento aquecedor não causa alguma explosão, uma vez que o sensor pode entrar em contato com gases inflamáveis, além de filtrar partículas suspensas e permitir que apenas gases adentrem na câmara. Já o elemento sensor, protegido pelo revestimento, é feito principalmente à base de dióxido de estanho (SnO_2), elemento este que é sensível aos gases a serem detectados, além de uma camada de óxido de alumínio que proporciona um aquecimento eficiente ao sensor (ENGINEERS, 2022). A Figura 9 demonstra este elemento do MQ-2.

Figura 9 – Elemento sensor do MQ-2



Fonte: ENGINEERS (2022).

Quando o elemento sensor está aquecido, moléculas de oxigênio são adsorvidas pela superfície de SnO_2 , fazendo com que os elétrons livres da banda de condução deste sejam atraídos pelo oxigênio e uma camada de depleção seja formada, criando uma barreira de potencial. Isto leva a camada de dióxido de estanho a possuir uma alta resistência e portando uma oposição à passagem de corrente elétrica. No entanto, quando gases redutores estão presentes, a quantidade de moléculas de oxigênio adsorvidas pela camada de SnO_2 é reduzida conforme este reage com os gases redutores, o que leva os elétrons a voltarem para a banda de condução, reduzindo assim a resistência do elemento sensor e permitindo a passagem de corrente (ENGINEERS, 2022).

3.2.2 YF-S201

O sensor YF-S201 é um sensor de fluxo de água composto por uma válvula de plástico, um rotor de água e um sensor de efeito Hall. A água flui através do rotor e este irá girar de acordo com o fluxo. Então o sensor Hall irá fornecer um sinal de pulso equivalente. Ou seja, esse sensor funciona através do efeito hall (LOUSADA, 2021).

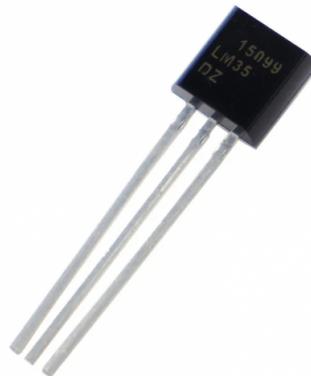
Figura 10 – Sensor YF-S201

Fonte: EASYTRONICS (2022).

3.2.3 LM35

O sensor de temperatura LM35 é um sensor de precisão com uma tensão de saída linear proporcional à temperatura quando alimentado com uma tensão de -20V DC e tem uma saída de sinal de 10mV para cada grau Celsius.

O LM35 não necessita de ajustes externos ou “trimming” para obter leituras de temperatura de forma precisa, com variações de 0,25°C a 0,75°C nas faixas de temperatura de -55°C a 150°C. Ele possui uma saída de baixa impedância, tensão linear e ajuste interno preciso, tornando as interfaces legíveis extremamente fáceis e o custo total do sistema significativamente menor (INSTRUMENTS, 2022).

Figura 11 – Sensor LM35

Fonte: BAUELETRÔNICA (2022).

3.3 DEFINIÇÃO DO MÉTODO DE COMUNICAÇÃO

Uma vez que ambos ESP32 e TTGO T-Call possuem integrados a tecnologia de comunicação Bluetooth Low Energy, optou-se pela utilização desta para comunicação entre microcontrolador periférico e central. Mais do que isso, o BLE se encaixa muito bem em aplicações IoT de escopo residencial, podendo alcançar distâncias de até 100m com linha de visada e taxa de dados de 1 a 3 Mbit/s (a taxa de dados efetiva do sistema acaba limitada pelo uso do 2G – o GPRS proporciona uma taxa máxima de 120 kbits/s), enquanto entrega um baixo consumo de energia, se alinhando muito bem também as necessidades da comunicação periférico-central, que envolve pequenas quantidades de dados e com períodos bem determinados (ARAUJO, 2012).

3.4 DESENVOLVIMENTO DO HARDWARE

O desenvolvimento do hardware foi dividido em três etapas:

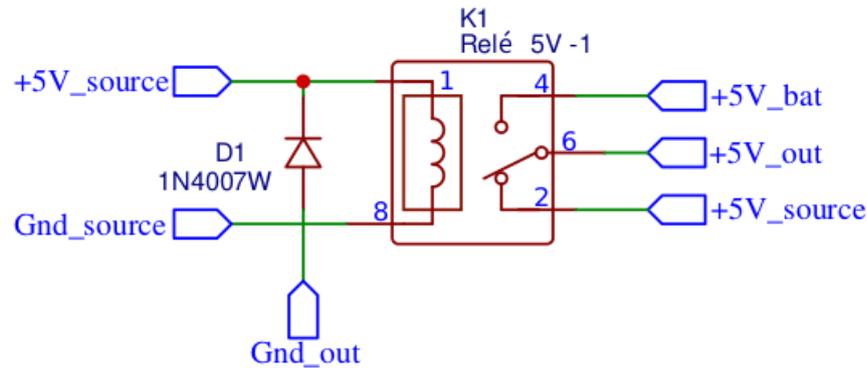
1. Circuito de alimentação;
2. Circuito do microcontrolador periférico;
3. Circuito do microcontrolador central

3.4.1 CIRCUITO DE ALIMENTAÇÃO

O circuito de alimentação é o mesmo a ser utilizado tanto no periférico quanto na central. Este circuito deve possuir a habilidade de alimentar sua carga utilizando a energia de uma fonte de tensão externa quando houver presença de energia nesta, ou chavear para alimentação através de uma bateria quando não houver. Sendo assim, foi escolhida a utilização de um relé de 5V, conectando a alimentação da bateria no contato normalmente fechado, e a alimentação da fonte no normalmente aberto, e a saída do circuito no comum, conforme Figura 12.

Como é utilizada uma bateria de 9V como fonte de energia alternativa, é necessário também o uso de um regulador de tensão para 5V, sendo escolhido o LM7805. Como este regulador consome uma corrente para manter-se ativo, de aproximadamente 3mA, foi adicionado outro relé de 5V para realizar o chaveamento quando há a presença de alimentação pela fonte externa, de maneira a não consumir a carga da bateria quando a energia da fonte está presente. Quanto a fonte externa, foi escolhida uma fonte AC/DC de 5V com capacidade mínima de corrente de 500mA (similar aos carregadores de celular), devido à facilidade de seu uso e baixo custo, podendo ser trocada facilmente em caso de falhas.

Figura 12 – Relé para circuito de alimentação



Fonte: Autoria própria (2023).

3.4.2 CIRCUITO DO PERIFÉRICO

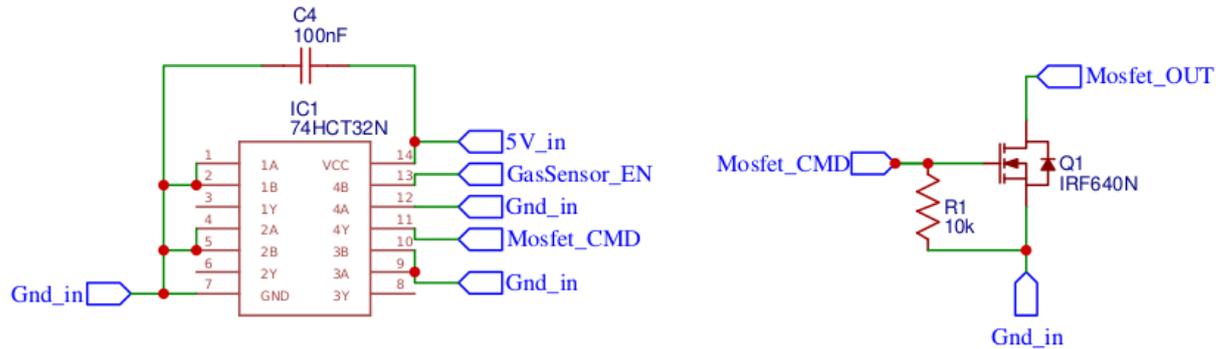
O hardware do periférico, além de seu microcontrolador - o ESP32 - necessita de blocos adicionais para a realização da aquisição de medidas da melhor maneira possível.

Para a utilização do sensor de gás MQ-2, viu-se a necessidade de um circuito de habilitação da alimentação deste, pois conforme seu datasheet a potência dissipada pode chegar até a 800mW, de maneira que a corrente medida foi de aproximadamente 120mA. Uma vez que o sistema pode ser alimentado por uma bateria, manter esse sensor sempre alimentado, além de desnecessário, se torna inviável uma vez que esgotaria a energia da bateria rapidamente. Sendo assim, foi desenvolvido um circuito de *enable* do sensor através de um mosfet canal N IRF640N, cujo comando vem de um pino de GPIO do ESP32. Para isso, foi necessário também a utilização de uma porta lógica OR 74HCT32 que atua como um conversor de nível lógico, uma vez que o ESP32 opera com 3.3V, tensão que não é suficiente para vencer a tensão V_{gs} de threshold do mosfet de maneira a saturá-lo e fechar a chave, cabendo ao 74HCT32 a conversão para 5V. Pelo mesmo motivo, foi preciso também um divisor resistivo no sinal de saída do sensor, que opera a 5V, para conectá-lo ao canal ADC do ESP32, conforme ilustram as Figuras 13 e 14.

3.4.3 CIRCUITO DA CENTRAL

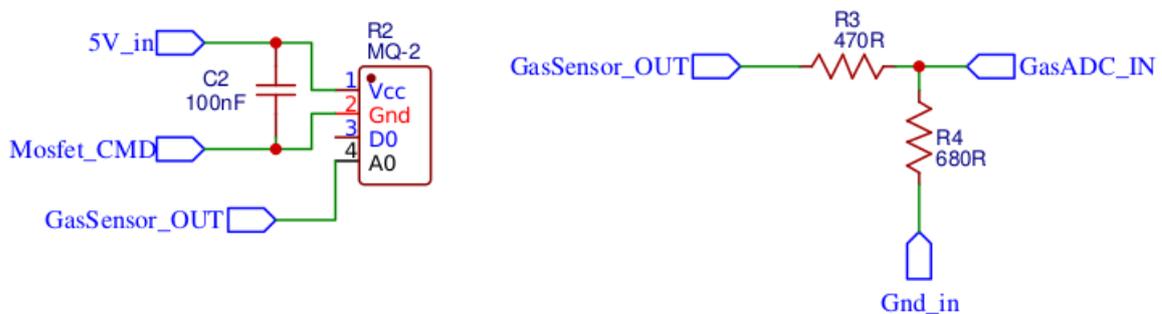
O hardware do microcontrolador central se demonstra mais simples que do periférico, pois conta com apenas o sensor de fluxo de água, o qual consome uma corrente relativamente baixa de aproximadamente 15mA, e não possui a possibilidade da realização de um circuito de *enable* como no sensor de gás, uma vez que envia seu sinal de medição de maneira assíncrona. Entretanto, foi inserido um led vermelho em seu circuito, de maneira que este acenda quando a central não realiza de maneira bem sucedida a conexão à APN do chip ou ao servidor da

Figura 13 – Circuito para adequação do sensor MQ-2 - a



Fonte: Autoria própria (2023).

Figura 14 – Circuito para adequação do sensor MQ-2 - b



Fonte: Autoria própria (2023).

aplicação. Por óbvio, é necessário também um chip nano de telefonia celular ativo e com pacote de dados disponível para inserção no slot do TTGO, sendo escolhido um chip da operadora CLARO para esta finalidade.

3.5 IMPLEMENTAÇÃO DO FIRMWARE

O firmware do microcontrolador periférico foi desenvolvido utilizando-se a linguagem C através das bibliotecas e API's disponibilizadas no ESP-IDF (Espressif IoT Development Framework) pela Espressif, fabricante do ESP32. Já no microcontrolador central, utilizou-se a linguagem C++, uma vez que foi utilizada a plataforma Arduino para escrita do código, devido ao grande suporte da comunidade e bibliotecas disponíveis para interação com o módulo GSM do TTGO T-Call. Para a programação de ambos microcontroladores foi utilizado o PlatformIO, uma extensão disponível no VSCode voltada para programação de microcontroladores e desenvolvimento de software embarcado, bem como foi utilizado o sistema operacional FreeRTOS, uma vez que este permite a criação de tasks com propósitos bem definidos, o que facilita a manutenção do código, e principalmente a temporização dos programas executados, o que é essencial

neste projeto uma vez que periférico e central se comunicam em intervalos de tempo específicos.

3.5.1 LÓGICA BÁSICA DO SISTEMA

Primeiramente, a central inicializa seus módulos (GSM, BLE, etc) e pinos de GPIO utilizados. Feito isso, ela tenta realizar a conexão à APN da rede de telefonia e se obtiver sucesso, realiza a conexão ao servidor da aplicação. Uma vez conectada no servidor, a central solicita (realiza um http GET) as configurações daquele dispositivo, isto é, as configurações de alertas e configurações de medidas. Por conveniência, no mesmo pacote http é inserido pelo servidor o unix time a ser utilizado pela central e periférico. Então a central, atuando como cliente BLE, passa a realizar o BLE scan procurando o periférico para se conectar, que atua como servidor BLE. A figura 12 exibe a tabela completa do serviço BLE do periférico. Ao se conectarem, o periférico, que realizava o advertising para ser descoberto pela central, pára de fazê-lo uma vez que a conexão já foi estabelecida. Logo após realizar a conexão, a central primeiro lê as características de medidas e estado dos alertas atuais do periférico, e então escreve nas características das configurações do periférico, aquelas que obteve do servidor, de maneira que o periférico atualiza e salva as configurações recebidas em arquivo do file system. Tal file system foi utilizado para garantir maior resiliência ao periférico, de maneira que em situações de reset ele possa simplesmente ler as configurações salvas em arquivo, além do estado dos alertas antes do dispositivo desligar. Após esse procedimento, a conexão BLE é encerrada, e a central envia para o servidor (http POST) as medidas e alertas atuais do sistema, isto é, medidas e alertas tanto do periférico quanto da própria central, e então se desconecta do servidor e desativa o módulo GSM. A partir deste momento, central e periférico voltam a se comunicar a cada X minutos, em que X é o tempo de coleta de medidas em minutos definido pelo usuário no aplicativo, de maneira que a task principal de cada um possa hibernar, otimizando o consumo de energia do sistema.

Uma vez que o projeto lida com medições que podem apresentar riscos à integridade da residência ou do usuário, foi pensado um “modo de supervisão” para o periférico. Este modo tem o intuito de reduzir drasticamente a frequência de coleta de medidas do periférico, quando este detecta que alguma medida está próxima do valor para geração do alerta, no entanto não o atingiu, sendo escolhida uma proporção de redução de 1:30 para redução da periodicidade das medições. Por exemplo, se a configuração para a coleta de medidas é de 10 minutos: $(10 \cdot 60) = 600$ segundos $\rightarrow 600 / 30 = 20$ segundos, em que este tempo é o tempo do supervisor. Além disso, enquanto estiver nesse modo, central e periférico permanecem com a conexão BLE ativa. Tal modo não é possível no microcontrolador central, uma vez que este obtém as medições de volume de maneira assíncrona.

Figura 15 – Tabela do serviço BLE do microcontrolador periférico

Serviço de monitoramento do periférico	
Característica do grupo de medidas	
Valor do grupo de medidas	
Configuração de notificação do grupo de medidas	
Característica dos alertas	
Valor dos alertas	
Configuração de notificação dos alertas	
Característica do estado do periférico	
Valor do estado	
Configuração de notificação do estado	
Característica das configurações de medidas	
Valor das configurações de medidas	
Característica das configurações de alertas	
Valor das configurações de alertas	
Característica do tempo do periférico	
Valor do tempo do periférico	

Fonte: Autoria própria (2023).

3.5.2 AQUISIÇÃO DAS MEDIDAS

A aquisição das medidas do sensor de gás e do sensor de temperatura foi feita através da utilização do ADC (*Analog to Digital Converter*), ou conversor analógico-digital. Este método consiste basicamente da amostragem do sinal analógico de entrada o qual é então quantizado, ou seja, discretizado com base na tensão de referência e da resolução do ADC do microcontrolador em questão. Tal quantização gera um erro de leitura que é no máximo metade da resolução do ADC. Desta forma quanto menor a resolução do ADC, isto é, quanto mais bits por Volts para quantização do sinal de entrada, menor o erro de quantização. (GUDINO, 2021)

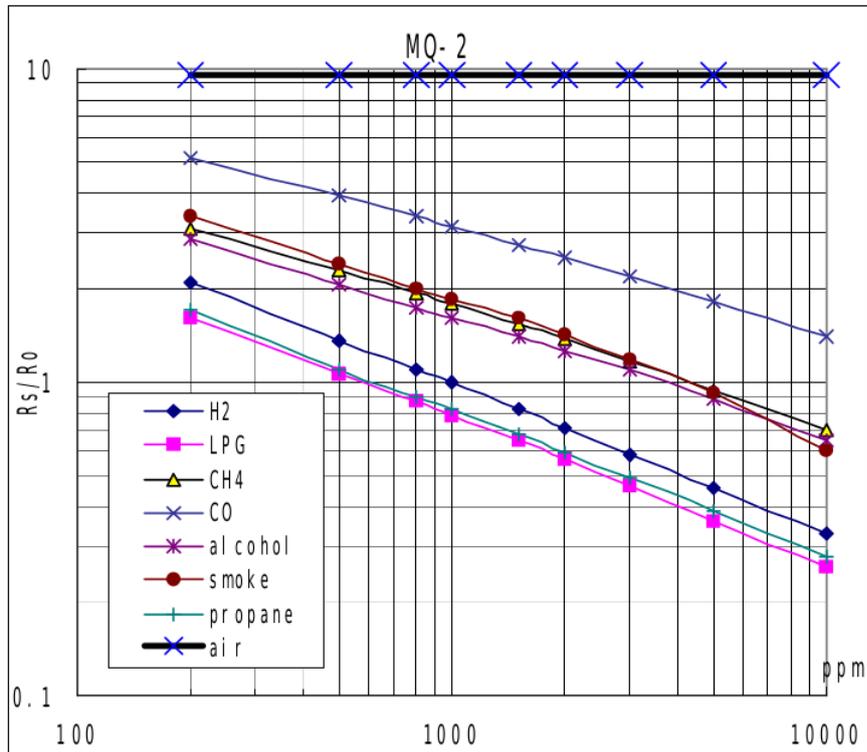
No caso do módulo ADC do ESP32, os sinais de entrada podem variar de 0 a 3,3V, com uma resolução de 12 bits, o que fornece um resultado de 0,8mV por degrau do sinal quantizado, e um erro menor que 0,4mV.

Como o sensor de gás é um resistor que varia sua resistência conforme a presença de gases, a qual está em série com uma resistência de proteção R_l de 5 kOhms, é necessário manipular a fórmula do divisor de tensão, uma vez que a tensão de saída do sensor de gás é em R_l , para se descobrir a resistência do sensor R_s . Onde V_{cc} é a tensão de alimentação do sensor, e V_l a tensão medida no resistor R_l , tem-se:

$$R_s = \frac{R_l * (V_{cc} - V_l)}{V_l}$$

Em posse dessa resistência, é possível aplicá-la a curva do sensor MQ-2 de maneira a descobrir a concentração de gás detectada pelo sensor. A Figura 16 demonstra a curva característica desse sensor para diferentes gases. Uma vez que o principal objetivo é a detecção de vazamentos de gás de cozinha, foi escolhida a curva do LPG como referência para o cálculo de concentração, porém deve-se notar que o sensor é sensível à todos gases listados no gráfico das curvas características.

Figura 16 – Curvas de resposta do sensor MQ-2



Fonte: ELETRONICS (2022).

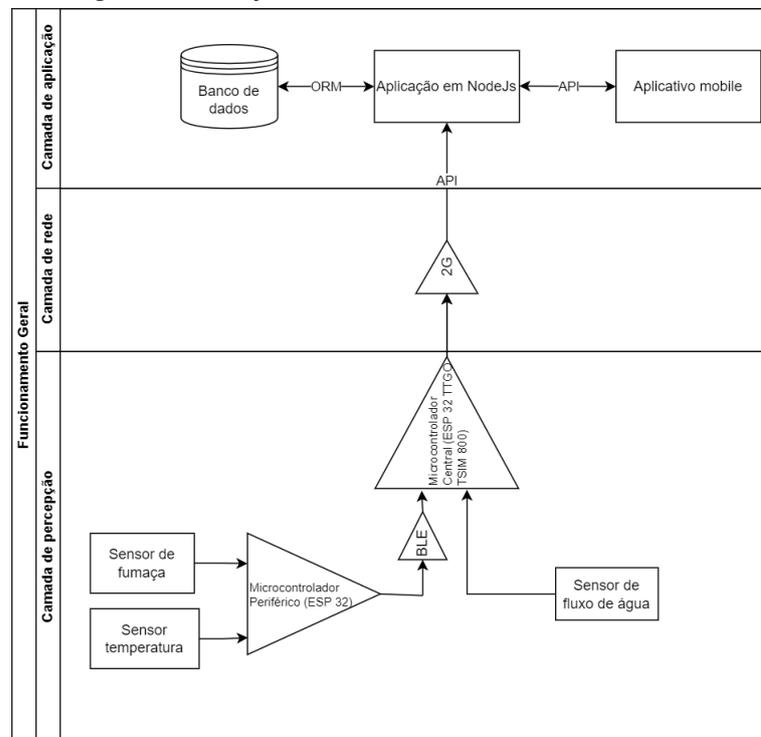
Para o sensor de temperatura LM35, basta aplicar a proporção de 10mV / °C para descobrir a temperatura ambiente.

Já o sensor de fluxo de água escolhido envia seu sinal de medição através de pulsos digitais, os quais possuem sua frequência diretamente proporcional à vazão detectada pelo sensor. Sendo assim, foi escolhida uma abordagem de leitura através de interrupção por borda de subida, a qual dispara uma task para realizar o processamento da leitura. Esta task então faz a contagem de quantos milissegundos o sinal ficou em nível alto, e na sequência em nível baixo, obtendo-se então o período e a frequência do sinal, de maneira que é possível obter a vazão instantânea e o volume de água que passou pelo cano naquele intervalo de tempo.

3.6 IMPLEMENTAÇÃO DO SERVIDOR

O servidor é o responsável pelos dados, seja no armazenamento, consulta e disponibilização dos mesmos para o usuário dentro do aplicativo mobile. Ele é composto por um banco de dados relacional e um código escrito em Java Script dentro do ambiente Node, que é responsável pela comunicação com o usuário por meio de requisições HTTP, que se comunicarão com o back end por meio de APIs. O serviço utilizado para a hospedagem da aplicação NodeJs e do banco de dados foi o da *Amazon Web Services (AWS)*. A figura 17 demonstra a arquitetura do sistema e como ela pode se encaixar no modelo de 3 camadas para IOT.

Figura 17 – Arquitetura do sistema em camadas IOT



Fonte: Autoria própria (2023).

3.6.1 BANCO DE DADOS

Para a parte de armazenamento do servidor, foi utilizado o banco de dados relacional MySQL, responsável por armazenar os dados de leitura dos sensores, as configurações de alertaS e medições e os alertas disparados. A figura 18 representa as tabelas dos banco de dados e suas relações.

A tabela “Users” representa os usuários que utilizaram o sistema, contendo as informações referente a ele. Ela é utilizada para relacionar os dados das demais tabelas a um usuário.

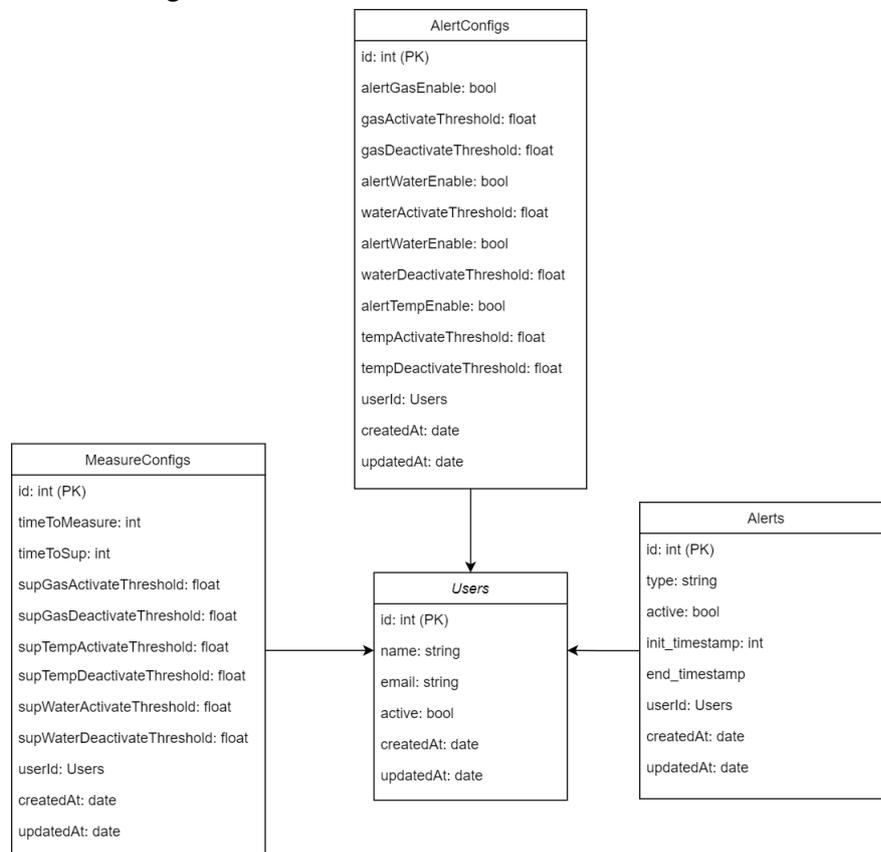
A tabela “Alerts” representa os alertas disparados pelo microcontrolador assim que uma medição estiver fora dos limites estabelecido pelo usuário. Contendo o tipo dele, se está ativo ou

não e o tempo de início e fim do mesmo. Ela é utilizada para a visualização dos alertas dentro do aplicativo mobile. Possui uma relação de muitos para um com a tabela “Users”.

A tabela “AlertConfigs” representa as configurações dos alertas, possuindo os limites de ativação que serão estabelecidos pelo usuário para cada tipo de sensor e se o mesmo ativo ou não. Ela é utilizada para o microcontrolador saber quando um alerta deverá ser disparado e para as configurações editáveis pelo usuário no aplicativo mobile. Possui uma relação de um para um com a tabela “Users”.

A tabela “MeasureConfigs” representa as configurações das medições, possuindo os limites do supervisor para cada sensor e o tempo de espera para aquisição das medidas no modo supervisor. Possui uma relação de um para um com a tabela “Users”.

Figura 18 – Modelo relacional do Banco de Dados



Fonte: Autoria própria (2023).

3.6.2 APLICAÇÃO NODEJS

Foi desenvolvida uma aplicação Javascript para coletar, tratar e salvar os dados recebidos do microcontrolador central e do aplicativo no banco de dados, assim como distribuir os mesmos a eles. Para execução do código desenvolvido foi utilizado o software NodeJS, que consiste em um interpretador Javascript capaz de ser executado diretamente pelo sistema ope-

racional sem a necessidade de um navegador. E para a comunicação com o banco de dados foi utilizado o ORM chamado Sequelize.

Para envio dos dados do servidor aos clientes, foi criada rotas de requisições do tipo “GET”. Em que o servidor retornará as configurações das medições e dos alertas por meio da rota “/configs”, e, também, através das rotas “/getLastMonthMed” e “/getLastHourMed”, o servidor retornará as medições do ultimo mês e da ultima hora respectivamente. Por meio da rota “/alerts” o servidor retornará os alertas disparados. Todos os dados retornados são em formato JSON e deve ser passado, em todas as requisições “GET”, dentro do parametro “query”, o identificador do usuário, por exemplo: “http://url/configs?userId=1”.

Para salvar os dados dentro do banco de dados, foi criada rotas de requisições do tipo “POST”. Em que o servidor armazenará as informações neste por meio dos dados recebidos pelo cliente dentro do parâmetro “body” em formato de JSON. Para o envio dos dados de leitura das medições dos sensores, foi utilizado a rota “/sendData” em que a aplicação validará as informações e salvará no banco de dados. Para o envio das configurações de medições e alertas, foram utilizadas as rotas “/measurement_configs” e “alert_configs” respectivamente, em que o servidor validará as informações e verificará se já existe uma configuração para o usuário mencionado, caso exista o servidor apenas atualizará a informação existente, caso contrário, o servidor criará uma nova configuração dentro do banco de dados para o usuário. Para o envio dos alertas disparados foi utilizado a rota “/alerts” em que a aplicação NodeJs validará os alertas e então salvará no banco de dados. Assim que o microcontrolador central realizar a requisição para salvar o alerta no banco de dados, o servidor enviará um email ao usuário, utilizando o protocolo SMTP, com o serviço do Gmail para disparo dos emails. Em todas as requisições “POST” é necessário passar o parametro “userId” para criar a relação dos dados com um usuário. Caso o mesmo não for encontrado, ou as informações não forem válidas, o servidor retornará um header 400, com uma mensagem de erro.

3.7 IMPLEMENTAÇÃO DO APLICATIVO

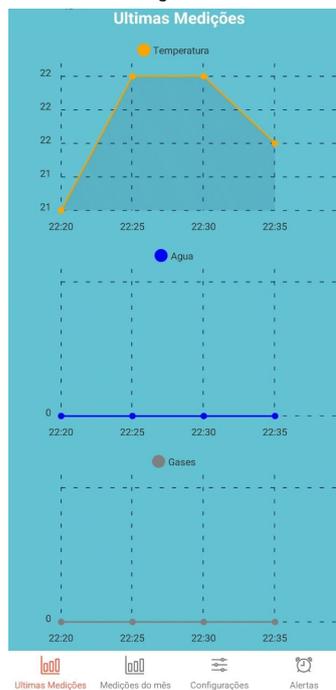
Para a disponibilização dos dados e interação com o usuário, foi desenvolvido um aplicativo mobile em JavaScript utilizando a biblioteca React Native. Dessa forma o usuário consegue acompanhar com maior facilidade as medições dos sensores instalados em sua residência e configurar os alertas sem precisar estar em sua casa. Para isso dividimos a aplicação em 4 telas: “Ultimas Medições”, “Medições do mês”, “Configurações” e “Alertas”.

3.7.1 ULTIMAS MEDIÇÕES

Nesta tela o usuário consegue monitorar os dados das medições dos sensores da ultima hora. A pagina é composta por três gráficos, um de temperatura, outro de água e um de gases.

Esses gráficos são alimentados por meio dos dados enviados pelo servidor, através de uma requisição “GET” que é feita automaticamente pelo aplicativo a cada 10 segundos, com intuito de manter os gráficos sempre atualizados, sem a necessidade de fechar e abrir o aplicativo o tempo todo. Nessa tela não há nenhuma interação com o usuário, servindo apenas para análise dos dados.

Figura 19 – Tela Últimas Medições dentro do aplicativo mobile



Fonte: Autoria própria (2023).

3.7.2 MEDIÇÕES DO MÊS

O funcionamento desta tela é igual a de “Últimas Medições”, com diferença apenas na rota que é feito o “GET” e na apresentação dos gráficos de temperatura, água e gases. Por meio da requisição o servidor retornará os dados das medições do mês, onde, para cada dia, é considerado a média das medições realizadas no mesmo.

3.7.3 CONFIGURAÇÕES

Nesta tela o usuário conseguirá editar as configurações referentes a leitura de cada medição, seja para temperatura, água ou gases, podendo ativar ou inativar os sensores e, também, aumentar ou diminuir os limiares para ativação e para desativação dos alertas. O utilizador poderá, também, alterar as configurações do modo de medição com supervisor, definindo uma porcentagem para conversão dos limiares, além do intervalo de tempo que as medições serão realizadas. O usuário poderá optar por salvar as configurações clicando no botão “Salvar” ou

Figura 20 – Tela Medições do mês dentro do aplicativo mobile



Fonte: Autoria própria (2023).

retornar as configurações padrões clicando no botão “Default” que aparecerão na tela. Ao clicar nesses botões, o aplicativo realizará uma requisição “POST” no servidor com as informações, para salva-las no banco de dados.

Figura 21 – Tela Configurações dentro do aplicativo mobile

Configurações do Sistema

Alerta de Temperatura
OFF / ON

Limiar para ativar: 50 °C
- +

Limiar para desativar: 45 °C
- +

Alerta de Água
OFF / ON

Limiar para ativar: 8 Litros
- +

Limiar para desativar: 4 Litros
- +

Alerta de Gases
OFF / ON

Limiar para ativar: 200 ppm
- +

Limiar para desativar: 100 ppm
- +

Fonte: Autoria própria (2023).

Fonte: Autoria própria (2023).

3.7.4 ALERTAS

A tela de “Alertas” tem como objetivo informar o usuário se existe algum alerta ativo ou inativo, e mostrar qual dia e o horário que foi realizado o disparo do mesmo e quando que foi finalizado para cada tipo de sensor.

Figura 22 – Tela Alertas dentro do aplicativo mobile



Fonte: Autoria própria (2023).

4 MÉTODOS E DISCUSSÃO DOS RESULTADOS

Este capítulo revisa os métodos de teste de todo o sistema desenvolvido e analisa os resultados obtidos. Os testes realizados foram divididos em testes das medições, testes do servidor e testes do aplicativo.

4.1 TESTES DE HARDWARE

De maneira a testar o circuito de alimentação do sistema, foi utilizado uma fonte de 5V via cabo USB e uma bateria de 9V, junto a um multímetro em paralelo com a tensão de saída para verificar a saída de 5V, esteja o circuito operando através da fonte, ou da bateria. Verificado isto, o multímetro foi posicionado em série com a alimentação da bateria para medir a corrente consumida por esta, e verificar que nenhuma corrente é drenada desta quando a alimentação da fonte está presente, o que ocorreu com sucesso.

Para o circuito do periférico, foi testado o bloco de enable do sensor MQ-2, verificando primeiramente a conversão da tensão de 3.3V na entrada para 5V de saída pelo 74HCT32N. Na sequência, foi verificada a saturação do mosfet e por consequência o fechamento do circuito de alimentação do sensor apenas quando há o sinal de tensão em nível alto entre o *Gate* e o *Source* do mesmo, o que também foi confirmado com sucesso.

4.2 TESTES DAS MEDIÇÕES

Para o teste das medições do sensor de gás foi utilizado um isqueiro para a emissão do gás butano, o qual é o principal componente do GLP junto ao propano, em que os resultados são apresentados na Tabela 2, e na Tabela 1 sem a emissão de gás pelo isqueiro. Por se tratar de um quimioresistor, o MQ-2 precisa passar por um breve período de poucos minutos de pré-aquecimento, o qual consiste de fornecer alimentação para o sensor em condições de ar limpo, e verificar que a concentração de gás fornecida pelo mesmo está estabilizada em quantidade próxima de zero. Para a calibração foi necessário ajustar o potenciômetro presente na placa do módulo, visto que existe uma resistência entra a saída do sensor e o terra, fator que determina o grau de sensibilidade do sensor, tomando como base as equações presentes no datasheet.

Tabela 1 – Medições com ar limpo

Multímetro (mV)	ADC (mV)	Erro (%)	Concentração (ppm)
200	178	11	24,33
178	158	11,24	16,19
178	168	5,61	20,10
171	153	10,53	14,35
165	148	10,30	12,60

Fonte: Autoria própria (2023).

Tabela 2 – Medições com presença de butano

Multímetro (mV)	ADC (mV)	Erro (%)	Concentração (ppm)
712	682	4,21	1095,99
602	571	5,15	682,57
773	751	2,85	1396,56
656	632	3,66	866,99
608	569	6,41	701,46

Fonte: Autoria própria (2023).

Para os testes de temperatura não foi feita calibração do sensor, uma vez que de acordo com o datasheet do LM35 o mesmo não requer nenhuma calibração, entregando temperaturas com acurácia de no mínimo 0,75°C. Iniciando a execução do teste, foi se aumentando gradativamente a temperatura no sensor, primeiro deixando-o normalmente no ambiente, e depois segurando-o para que tivesse sua temperatura aumentada, e para as temperaturas mais elevadas através de um ferro de solda, pressionado por poucos segundos contra o sensor.

Tabela 3 – Medições de temperatura

Multímetro (mV)	ADC (mV)	Erro (%)	Temperatura (°C)
271	249	8,12	24,9
290	270	6,90	27,0
308	285	7,47	28,5
322	300	6,83	30,0
359	338	5,85	33,8
387	371	4,13	37,1
414	400	3,38	40,0
453	438	3,31	43,8
490	471	3,88	47,1
516	498	3,49	49,8

Fonte: Autoria própria (2023).

Para a medição do volume de água detectado através do sensor de fluxo, na falta de um balde graduado, foi utilizada uma jarra com uma graduação em volumes de 250mL, conforme demonstram as figuras 21 e 22. O sensor utilizado não apresenta informações de calibração no seu datasheet.

Tabela 4 – Medições de volume de água

Volume na jarra (mL)	Medição 1 (mL)	Medição 2 (mL)	Medição 3 (mL)	Medição 4 (mL)	Medição 5 (mL)	Média (mL)	Erro (%)
250	120	110	110	110	120	114	54,40
500	330	320	310	320	320	320	36,00
750	660	660	670	650	670	662	11,73
1000	940	930	940	920	930	932	6,80
1250	1210	1200	1220	1200	1200	1206	3,52

Fonte: Autoria própria (2023).

Nota-se o erro considerável obtido para as medições de 250mL e de 500mL. Isto se explica devido ao fato de que para atingir com precisão pequenos volumes na jarra, foi necessário deixar a torneira utilizada à uma vazão baixa, que certamente está fora da faixa de operação

Figura 23 – Jarra utilizada para as medições de água



Fonte: Aatoria própria (2023).

Figura 24 – Configuração utilizada para as medições de água



Fonte: Aatoria própria (2023).

ideal do sensor (que conforme datasheet é de 100 l/h a 1800 l/h, ou seja, de 1,67 a 30 l/min). Entretanto, observa-se que para volumes maiores e, portanto, a utilização de uma vazão maior na torneira, o erro atingiu porcentagens bastante satisfatórias, chegando a apenas 3,52% para 1250mL, uma vez que a quantidade exata do volume não é o principal objetivo do sistema, embora se espere alguma exatidão.

4.3 TESTE DA COMUNICAÇÃO PERIFÉRICO-CENTRAL

Para testar a comunicação BLE entre central e periférico, ambos foram posicionados a diferentes distâncias e em diferentes cômodos. Foi verificado que a distância máxima de alcance para uma comunicação bem-sucedida foi de 10m sem linha de visada, e de 30m com linha de visada, o que se demonstra uma distância aceitável para pequenas a médias residências, a depender da distância entre caixa d'água, onde a central deve ser posicionada, e a cozinha, local recomendado para posicionamento do periférico.

4.4 TESTE DA COMUNICAÇÃO 2G DA CENTRAL

A fim de efetuar o teste da comunicação 2G do microcontrolador central, este foi posicionado em diferentes cômodos e em diferentes posições. Naturalmente, algumas localizações se demonstraram mais suscetíveis a uma conexão bem-sucedida à APN do que outras, e algumas realizando a conexão mais rapidamente e outras mais lentamente. Porém, de maneira geral a central se conectou e realizou a comunicação com o servidor com sucesso na maioria das localizações em que foi testada, sendo os cômodos do andar mais baixo da residência em que foi feito o teste os que apresentaram mais dificuldade de conexão, devido a um RSSI mais fraco.

4.5 TESTES DO SERVIDOR

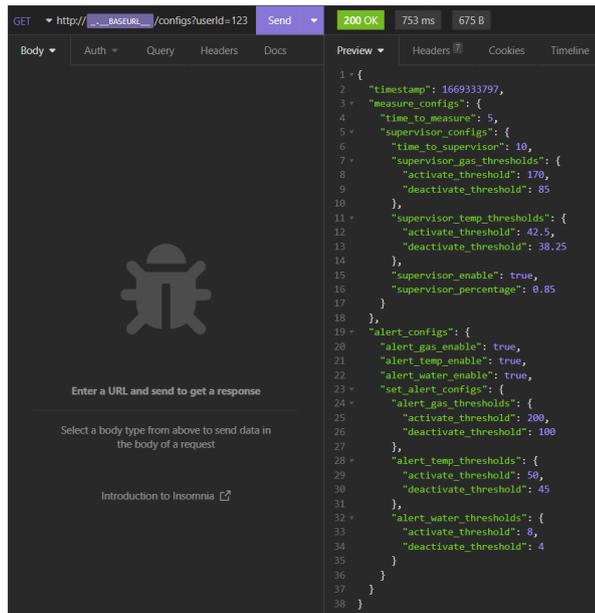
Para os testes e validação das requisições do servidor, foi utilizado o programa chamado Insomnia, uma ferramenta dedicada para desenvolvimento e teste de API. Dessa forma é possível configurar as informações a serem enviadas na requisição através dos parâmetros "body", "query" e "header" e obter a resposta detalhada do servidor. Facilitando a verificação e adequação da resposta em JSON.

Para os testes e verificação dos dados gravados no banco de dados foi utilizado o Bee-Keeper, uma ferramenta cujo objetivo é auxiliar na visualização da estrutura da base de dados e para realizar pequenos ajustes nas tabelas como inserção e remoção de dados. Dessa forma foi possível validar o que está sendo gravado e simular dados para testes.

4.6 TESTES DO APLICATIVO

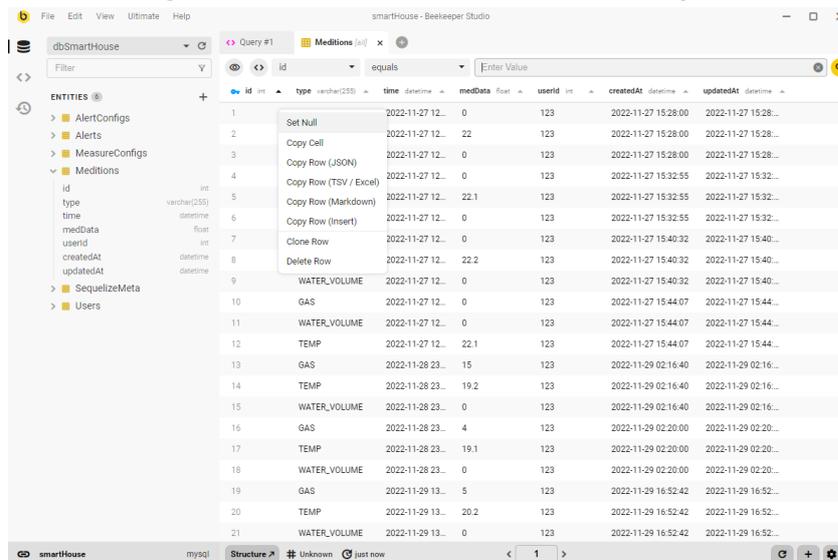
Para os teste do aplicativo foi utilizado a ferramenta Nox Player, que consiste em um simulador de celular com sistema operacional Android. Integrando ele com o React Native, foi possível realizar testes em tempo real, de forma que uma alteração no código refletia, dentro de poucos segundos, na alteração do aplicativo dentro do simulador, sem a necessidade de reiniciar a aplicação a cada modificação.

Figura 25 – Requisição GET feita na ferramenta insomnia para o servidor e seu retorno em JSON



Fonte: Autoria própria (2023).

Figura 26 – Interface da ferramenta BeeKeeper



Fonte: Autoria própria (2023).

4.7 TESTE DA GERAÇÃO DE ALERTAS

Para este teste, foram configurados pelo aplicativo thresholds de ativação de alerta baixos de maneira a facilitar a geração de alertas, sendo 200 ppm para o gás, 25°C para temperatura e 2L para volume de água. Realizando os mesmos procedimentos descritos na subseção TESTES DAS MEDIÇÕES, conseguiu-se atingir os thresholds de ativação de cada tipo de medida e verificar a geração do alerta na interface do aplicativo, com o estado de cada alerta ativo e o horário de início apropriado, assim como o envio de e-mail para o usuário. De forma similar, foi verificada a desativação dos alertas quando as medidas encontraram-se abaixo do thresholds

Figura 27 – Aplicativo aberto no simulador Nox



Fonte: Autoria própria (2023).

de desativação, trocando o estado dos alertas para desativo e registrando o horário de fim da ocorrência.

4.8 ANÁLISE DE CUSTOS

A Tabela 4 apresenta uma análise detalhada dos custos para produção do kit de monitoramento, sensores e componentes eletrônicos utilizados no protótipo.

Tabela 5 – Custos para desenvolvimento

Componente	Quantidade	Preço (R\$)
ESP 32	1	50,00
Ttgo T-call Sim800l	1	150,00
LM35 (Sensor de Temperatura)	1	15,00
MQ2 (Sensor de gases)	1	20,00
YF-S201 (Sensor de Fluxo de água)	1	35,00
Chip 2G	1	10,00
Relés	4	20,00
PCB Módulo Central	1	5,00
PCB Módulo Periférico	1	5,00
Total		310,00

Fonte: Autoria própria (2023).

5 CONSIDERAÇÕES FINAIS

Finalizado o processo de desenvolvimento do sistema, podemos destacar que o mesmo pode ser instalado apenas em residências que possuam conexão de rede celular, visto que a central utiliza da internet 2G para se comunicar com o servidor. E, também, será necessário realizar recargas no chip telefônico para que o mesmo sempre tenha dados disponíveis para as requisições HTTP. Outro ponto seria a questão da comunicação entre o microcontrolador periférico e central, de modo que ambos podem ficar a uma distância máxima de aproximadamente 10 metros. Ou seja, a caixa d'água e os sensores de temperatura e gases não podem ficar muito distantes uns dos outros.

Através da análise das medições dos sensores, foi possível concluir que o sensor de fluxo de água foi o que apresentou uma média porcentual de erro maior em comparação com as demais, principalmente quando o volume de água é inferior a 1000 mL. Enquanto que o sensor de temperatura foi o que apresentou um porcentual de erro menor. Quanto a comunicação da central com o servidor, foi constatada uma considerável latência nas requisições HTTP, visto que é utilizada a internet 2G para comunicação.

Dentre todas as opções semelhantes encontradas no mercado, como o sensor HISSGL da empresa Geonav e o IDG 620 da Intelbras, que custam em torno de R\$ 250,00 e R\$ 600,00 (detector de gás + hub de automação smart) respectivamente, mencionados na sessão 1.3 do trabalho, é possível observar que os produtos já disponíveis contam apenas com um sistema de detecção de gases inflamáveis, ou com apenas com a detecção de aumento de temperatura e outros apenas com a detecção de vazamento de água. Enquanto que a solução proposta oferece uma maior segurança ao usuário, visto que detecta as 3 possíveis ameaças e ainda proporciona uma melhor experiência ao cliente com um preço um pouco superior ao sensor HISSGL e inferior ao IDG 620. Uma vez que o mesmo consegue monitorar as medições, alertas e configurar o sistema por meio de seu celular. Com um custo médio de R\$ 310,00 levando em conta todos os sensores, microcontroladores e componentes utilizados no protótipo.

Nesse sentido, a utilização do sistema garante ao usuário um maior conforto e segurança ao auxiliá-lo na gestão de suas residências. Além disso, pode gerar uma grande economia em casos de incêndios e vazamento de água.

5.1 IMPLEMENTAÇÕES FUTURAS

Com o objetivo de aprimorar e dar continuidade ao trabalho, seria possível, na parte do servidor, amplificar a segurança das requisições HTTP utilizando criptografia assimétrica, onde o servidor possuiria a chave privada e os agentes (aplicativo mobile e o microcontrolador central) a chave pública, possibilitando a comunicação apenas entre eles e garantindo, assim, uma maior confiabilidade do sistema. Também seria interessante utilizar a plataforma disponibilizada pelo Google chamada firebase, visto que ela permite enviar mensagens do servidor ao aplicativo

por meio de "push notifications", para que, dessa forma, o usuário consiga tratar o incidente de forma mais rápida e eficaz. Já na comunicação entre microcontrolador periférico e central, uma melhoria significativa seria a implementação de segurança no BLE, através do emparelhamento, ligação (*bonding*) e criptografia, garantindo a confidencialidade e integridade das informações trocadas por ambos.

REFERÊNCIAS

- AERP. **Mais de 300 ocorrências de vazamento de gás de cozinha são registradas no Paraná.** 2022. Disponível em: <https://aerp.org.br/redeaerp/mais-de-300-ocorrencias-de-vazamento-de-gas-de-cozinha-sao-registradas-no-parana/>. Acesso em: 11 out. 2022.
- ALIBABABA. **ESP32 ESP-32S Placa de Desenvolvimento Wi-fi ESP-WROOM-32 NodeMCU-32S ESP32 Processador Chip Integrado CP2102 para Arduino Microcontrolador.** 2022. Disponível em: <https://portuguese.alibaba.com/product-detail/ESP32-ESP-32S-WiFi-Development-Board-1600129565598.html>. Acesso em: 15 nov. 2022.
- ALIEXPRESS. **Lilygo ttgo módulo t-call v1.3, esp32 módulo sem fio, antena sim, módulo sim800l e antena gsm/gprs.** 2022. Disponível em: <https://pt.aliexpress.com/item/1005002035191360.html>. Acesso em: 15 nov. 2022.
- ALURA. **Node.JS: definição, características, vantagens e usos possíveis.** 2022a. Disponível em: <https://www.alura.com.br/artigos/node-js-definicao-caracteristicas-vantagens-usos>. Acesso em: 13 nov. 2022.
- ALURA. **React Native: o que é e tudo sobre o Framework.** 2022b. Disponível em: <https://www.alura.com.br/artigos/react-native>. Acesso em: 12 nov. 2022.
- AMAZON. **Hi by Geonav Sensor Inteligente para detecção de Gás Natural e GLP, Wi-Fi, Aplicativo, Bivolt, HISSGL.** 2022. Disponível em: <https://www.amazon.com.br/Geonav-Inteligente-detec%3%7%3%3o-Aplicativo-HISSGL/dp/B0828KVX7N>. Acesso em: 12 nov. 2022.
- ANTUNES, P. I. T. D. C. **Medição de posição de rotor em mancal magnético através de sensor hall.** 2012. Disponível em: https://www.teses.usp.br/teses/disponiveis/3/3152/tde-10062013-152331/publico/dissertacao_Pedro.pdf. Acesso em: 15 nov. 2022.
- AWS. **O que é a Internet das Coisas (IoT)?** 2022a. Disponível em: <https://aws.amazon.com/pt/what-is/iot/#:~:text=O%0termo%0IoT%C%0ou%0Internet,como%0entre%0os%0pr%3%3prios%0dispositivos>. Acesso em: 20 out. 2022.
- AWS. **O que é uma API?** 2022b. Disponível em: <https://aws.amazon.com/pt/what-is/api/>. Acesso em: 21 nov. 2022.
- BAUELETRÔNICA. **Sensor de Temperatura LM35.** 2022. Disponível em: <https://www.baudaeletronica.com.br/sensor-de-temperatura-lm35.html>. Acesso em: 01 dez. 2022.
- BECKER, L. **O que é React Native?** 2021. Disponível em: <https://www.organicadigital.com/blog/o-que-e-react-native/>. Acesso em: 06 out. 2022.
- BERTULANI, C. A. **O Efeito Hall Quântico.** 2005. Disponível em: <https://web.archive.org/web/20051102111237/http://www.if.ufrj.br/teaching/hall/hall.html>. Acesso em: 25 nov. 2022.
- CBMCE. **Corpo de Bombeiros do Ceará registrou 1.650 ocorrências com gases em 2022.** 2022. Disponível em: <https://www.bombeiros.ce.gov.br/2022/10/16/corpo-de-bombeiros-do-ceara-registrou-1-650-ocorrencias-com-gases-em-2022/>. Acesso em: 11 out. 2022.

- DEV MEDIA. **ORM : Object Relational Mapper**. 2011. Disponível em: <https://www.devmedia.com.br/orm-object-relational-mapper/19056>. Acesso em: 21 nov. 2022.
- DW. **Mesmo com crise hídrica, Brasil perde 40% da água tratada**. 2022. Disponível em: <https://www.dw.com/pt-br/mesmo-com-crise-h%C3%A9-brasil-perde-40-da-%3A1gua-tratada/a-61208282#:~:text=Crescente%0afalta%0de%0%3A1gua,n%3A3o%0acabou%C%0segundo%0Adriana%0Cuartas>. Acesso em: 11 out. 2022.
- EASYTRONICS. **YFS201 Sensor de Vazão 30L/min | YF-S201**. 2022. Disponível em: <https://www.easytronics.com.br/sensor-de-vazao-yfs201>. Acesso em: 11 dez. 2022.
- ELETRONICS, H. **TECHNICAL DATA MQ-2 GAS SENSOR**. 2022. Disponível em: <https://www.mouser.com/datasheet/2/321/605-00008-MQ-2-Datasheet-370464.pdf>. Acesso em: 27 nov. 2022.
- ENGINEERS, L. M. **How MQ2 Gas/Smoke Sensor Works? Interface it with Arduino**. 2022. Disponível em: <https://lastminuteengineers.com/mq2-gas-senser-arduino-tutorial/>. Acesso em: 19 nov. 2022.
- FOGAÇA, J. **Composição do gás de cozinha**. 2022. Disponível em: <https://brasilescola.uol.com.br/quimica/composicao-gas-cozinha.htm>. Acesso em: 11 nov. 2022.
- GLOBO. **Morador sofre queimaduras em 60% do corpo em explosão na cozinha de apartamento em Assis**. 2022. Disponível em: <https://g1.globo.com/sp/bauru-marilia/noticia/2022/05/02/morador-sofre-queimaduras-em-60percent-do-corpo-em-explosao-na-cozinha-de-apartamento-em-assis.gh.html>. Acesso em: 11 out. 2022.
- GOUVEIA, R. **Medidas de Volume**. 2022. Disponível em: <https://www.todamateria.com.br/medidas-de-volume/>. Acesso em: 11 nov. 2022.
- INSTRUMENTS, T. **LM35 Precision Centigrade Temperature Sensors**. 2022. Disponível em: <https://www.ti.com/lit/ds/symlink/lm35.pdf>. Acesso em: 14 nov. 2022.
- KITAMURA, C. **O Que É Protocolo De Comunicação?** 2022. Disponível em: <https://celsokitamura.com.br/o-que-e-protocolo-de-comunicacao/>. Acesso em: 21 nov. 2022.
- LONDRINA, F. D. **Reajuste de 4,96% na tarifa da Sanepar entra em vigor nesta terça**. 2022. Disponível em: <https://www.folhadelondrina.com.br/economia/reajuste-de-496-na-tarifa-da-sanepar-entra-em-vigor-nesta-terca-3195431e.html>. Acesso em: 22 out. 2022.
- LOUSADA, R. **Guia Prático do Sensor de Fluxo de Água**. 2021. Disponível em: <https://blog.eletrogate.com/sensor-de-fluxo-de-agua/>. Acesso em: 21 nov. 2022.
- MAMUTELETRÔNICA. **Módulo Sensor de Gás MQ-2**. 2022. Disponível em: <https://www.mamuteletronica.com.br/modulo-sensor-de-gas-mq-1280>. Acesso em: 20 nov. 2022.
- MDN. **Uma visão geral do HTTP**. 2022. Disponível em: <https://developer.mozilla.org/pt-BR/docs/Web/HTTP/Overview>. Acesso em: 23 nov. 2022.
- NAIK, N. Choice of effective messaging protocols for iot systems: Mqtt, coap, amqp and http. **2017 IEEE International Systems Engineering Symposium (ISSE)**, p. 1–7, 2017.
- NÓBREGA, A. **Fumaça: o que é e como se proteger**. 2022. Disponível em: <https://www.ecycle.com.br/fumaca/>. Acesso em: 11 nov. 2022.

ORACLE. **O que é um Banco de Dados?** 2022. Disponível em: <https://www.oracle.com/br/database/what-is-database/>. Acesso em: 17 nov. 2022.

POSTEL, J. B. **SIMPLE MAIL TRANSFER PROTOCOL**. Marina del Rey, California, 1982.

RIABOV, V. V. **SMTP (Simple Mail Transfer Protocol)**. Nashua, New Hampshire, 2005.

ROCHA, R. D. C. **Projeto De Controlador De Temperatura Para Mantas Termoelétricas**. 2012. Disponível em: <http://lyceumonline.usf.edu.br/salavirtual/documentos/2339.pdf>. Acesso em: 14 nov. 2022.

ROCKETSEAT. **Tipos de Parâmetros nas requisições REST**. 2020. Disponível em: <https://blog.rocketseat.com.br/tipos-de-parametros-nas-requisicoes-rest/>. Acesso em: 23 nov. 2022.

SHADRACK, B. **HTTP Methods "GET", "POST", "PUT", "PATCH", "DELETE"**. 2022. Disponível em: <https://dev.to/qbentil/http-methods-get-post-put-patch-delete-1fhi>. Acesso em: 23 nov. 2022.

SILAS, J. **O que é temperatura?** 2022. Disponível em: <https://brasilecola.uol.com.br/o-que-e/fisica/o-que-e-temperatura.htm>. Acesso em: 11 nov. 2022.

TECHTUDO. **O que é e como usar o MySQL?** 2012. Disponível em: <https://www.techtudo.com.br/noticias/2012/04/o-que-e-e-como-usar-o-mysql.ghtml>. Acesso em: 21 nov. 2022.

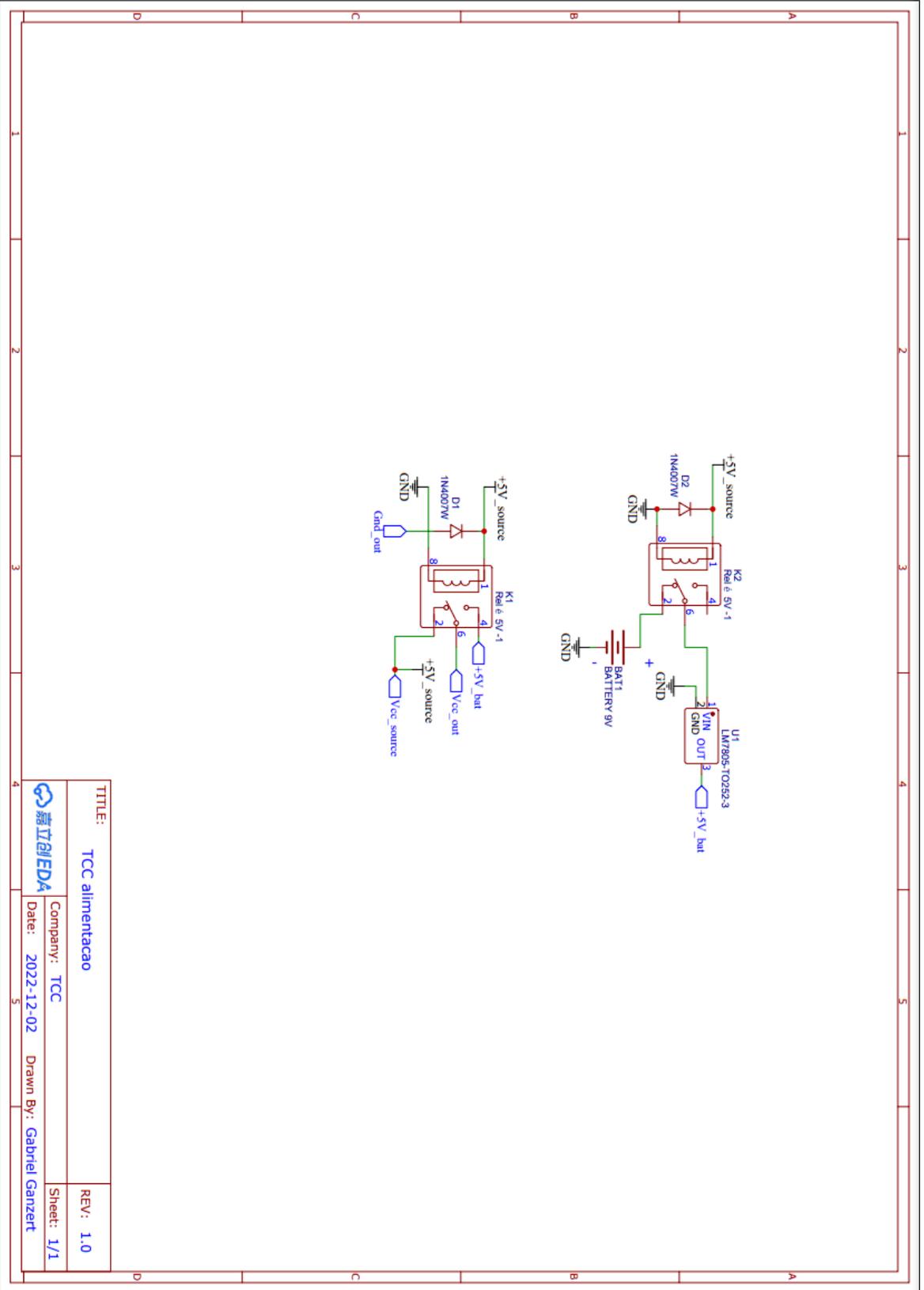
TECNOBLOG. **O que é Node.js? [Guia para iniciantes]**. 2020. Disponível em: <https://tecnoblog.net/responde/o-que-e-node-js-guia-para-iniciantes/>. Acesso em: 13 nov. 2022.

TEKADE, V. **Getting Started with Backend Development**. 2021. Disponível em: <https://medium.com/coders-capsule/getting-started-with-backend-development-8ce55585e860>. Acesso em: 08 nov. 2022.

TREINAWEB. **O que é ORM?** 2019. Disponível em: <https://www.treinaweb.com.br/blog/o-que-e-orm>. Acesso em: 21 nov. 2022.

APÊNDICE A – ESQUEMÁTICO DO HARDWARE DO PERIFÉRICO

APÊNDICE B – ESQUEMÁTICO DO CIRCUITO DE ALIMENTAÇÃO



TITLE:	TCC alimentacao	REV:	1.0
Company:	TCC	Sheet:	1/1
Date:	2022-12-02	Drawn By:	Gabriel Ganzert



APÊNDICE C – ESQUEMÁTICO DO HARDWARE DA CENTRAL

