



UNIVERSIDADE FEDERAL DO PARÁ
CAMPUS UNIVERSITÁRIO DE TUCURUÍ
FACULDADE DE ENGENHARIA DA COMPUTAÇÃO

VICTOR VINÍCIUS EUSTÁQUIO DE ALMEIDA

SISTEMA DE CONTROLE E MONITORAMENTO DE VAZÃO

TUCURUÍ/PA
2025



UNIVERSIDADE FEDERAL DO PARÁ
CAMPUS UNIVERSITÁRIO DE TUCURUÍ
FACULDADE DE ENGENHARIA DE COMPUTAÇÃO

TÍTULO: SISTEMA DE CONTROLE E MONITORAMENTO DE VAZÃO.

DISCENTE: Victor Vinícius Eustáquio de Almeida

MATRÍCULA: 201433840046

#	BANCA EXAMINADORA	CONDIÇÃO
1	<i>Prof. Dr. Daniel da Conceição Pinheiro</i>	<i>Orientador</i>
2	<i>Prof. Dr. Marco José de Sousa</i>	<i>Membro interno</i>
3	<i>Prof. Dr. Renato Luz Cavalcante</i>	<i>Membro interno</i>

Data da Defesa: 11/03/2025 | **Hora Início:** 14:15 | **Hora Término:** 16:20

Trabalho Escrito (0 a 10 pontos por critério)	Examinador 1	Examinador 2	Examinador 3
Formatação	9,50	9,00	9
Linguagem (gramática e semântica)	8,50	9,00	9
Conteúdo técnico	9,20	9,50	9
Defesa Oral (0 a 10 pontos por critério)	Examinador 1	Examinador 2	Examinador 3
Sequência lógica de apresentação	9,00	9,00	9
Administração do tempo	10,00	10,00	9
Expressão oral	8,00	8,50	9
Domínio do tema	9,50	10,00	9
Média por examinador	9,10	9,29	9,00
Média Final	9,13		
Conceito Final	EXC		

Tucuruí-Pa, 11 de março de 2025.

Documento assinado digitalmente
gov.br DANIEL DA CONCEICAO PINHEIRO
Data: 11/03/2025 16:25:27-0300
Verifique em <https://validar.iti.gov.br>

Documento assinado digitalmente
gov.br RENATO LUZ CAVALCANTE
Data: 11/03/2025 17:06:24-0300
Verifique em <https://validar.iti.gov.br>

Orientador

Documento assinado digitalmente
gov.br MARCO JOSE DE SOUSA ROCHA
Data: 11/03/2025 17:00:27-0300
Verifique em <https://validar.iti.gov.br>

Membro interno

Membro interno

Membro interno

VICTOR VINÍCIUS EUSTÁQUIO DE ALMEIDA

SISTEMA DE CONTROLE E MONITORAMENTO DE VAZÃO

Trabalho de Conclusão de Curso, apresentado como requisito parcial para a obtenção de grau de Bacharel em Engenharia de Computação, pela Universidade Federal do Pará.

Orientador: Prof. Dr. Daniel da Conceição Pinheiro
Universidade Federal do Pará

TUCURUÍ/PA
2025

Dedico este trabalho a Maria do Carmo Barbosa de Almeida, matriarca de nossa família, e às suas filhas, Almiria Barbosa de Almeida e Luzimeres Barbosa de Almeida, que tenho a honra de chamar de mães. Sem a dedicação e o apoio de vocês, esta conquista não seria possível.

AGRADECIMENTOS

Gostaria de expressar minha profunda gratidão ao meu orientador, Prof. Dr. Daniel da Conceição Pinheiro, pela sua inestimável orientação, apoio e incentivo durante todo o processo de elaboração deste TCC. Agradeço a paciência, as valiosas sugestões e a constante disponibilidade para me auxiliar na superação dos desafios encontrados. Sem a sua expertise e dedicação, este trabalho não seria possível.

Agradeço também aos meus colegas de curso, Dárlison Souza de Alencar, Bruno Braga Pinheiro, Gabriel Ventura Giordano, pelas valiosas discussões e colaboração durante a pesquisa. Aos colegas de estágio na Vale S.A, Roberto Thomé da Rocha, Gustavo Henrique dos Santos Torres, Lorrann Borges, Dílson Júnior, meu sincero agradecimento pela troca de experiências ao longo desta jornada. Compartilhar desafios e conquistas com vocês tornou esse período mais enriquecedor e gratificante. Cada aprendizado e momento vivido contribuiu imensamente para o meu crescimento profissional e pessoal. Obrigado por essa caminhada juntos!

Agradeço ao memorável amigo Janderson Tomé de Souza, cuja amizade, generosidade e força sempre foram inspiração para mim. Sua partida, vítima da COVID-19, deixou uma lacuna imensurável, mas seu legado seguirá vivo em nossas lembranças e em tudo que realizamos. Esta conquista é, em parte, também sua.

Sou grato à minha família e amigos pelo apoio incondicional durante toda a minha trajetória acadêmica. Agradeço minha segunda mãe, Almiria Barbosa de Almeida, minha profunda gratidão por todo amor, dedicação, ensinamentos e cobranças. Como mãe, foi meu alicerce; como educadora, guiou-me pelo caminho do conhecimento e dos valores que carrego comigo. Seu exemplo de sabedoria e generosidade moldou quem sou hoje. Esta conquista é, em grande parte, reflexo do que aprendi com você. Obrigado por tudo!

À minha querida avó, Maria do Carmo Barbosa de Almeida, que sempre perguntava quando o curso acabaria para que eu pudesse regressar à minha cidade natal e voltar a visitá-la com mais frequência, expressei meu eterno carinho e gratidão. Sua preocupação e seu amor incondicional sempre foram um refúgio para mim. Mesmo não estando mais entre nós, levo comigo suas palavras, seus ensinamentos e o aconchego do seu afeto. Esta conquista também é sua, e onde quer que esteja, espero que se orgulhe de mim.

Finalmente, agradeço à Universidade Federal do Pará por me proporcionar a oportunidade de realizar este trabalho de conclusão de curso. Agradeço especialmente ao Centro Acadêmico de Engenharia da Computação, assim como ao Sérgio Pozzebon Júnior e ao Josinaldo Machado da Silva, pelo acesso aos recursos e à infraestrutura necessários para o desenvolvimento deste projeto.

“A maior recompensa para o trabalho do homem não é o que ele ganha com isso, mas o que ele se torna com isso.” (John Ruskin)

RESUMO

Esta monografia trata do desenvolvimento de um sistema de controle e monitoramento de vazão para a gestão eficiente do consumo de água em banheiros, utilizando tecnologias de sistemas embarcados. O principal objetivo é desenvolver um sistema capaz de acionar a água em mictórios de forma autônoma e monitorar, em tempo real, a vazão, o que resulta em economia financeira e na redução dos impactos ambientais ao limitar o desperdício hídrico. O sistema foi desenvolvido com sensores de presença, sensor de fluxo, microcontrolador (*NodeMCU ESP32*), relé de controle e válvula solenoide. A programação do hardware foi realizada na plataforma de desenvolvimento *Arduino*. Esse hardware opera em conjunto com um banco de dados *PostgreSQL* e um servidor web em *Node.js*. O diferencial do sistema está na abrangência das tecnologias utilizadas, que incluem desde sistemas embarcados para a coleta de dados até uma plataforma web com interface para visualização, controle e análise das informações.

Palavras-chave: Sustentabilidade; Monitoramento; Controle; Sistemas Web; Internet das Coisas;

ABSTRACT

This paper focuses on the development of a flow control and monitoring system for efficient water consumption management in restrooms, using embedded system technologies. The main objective is to develop a system capable of autonomously activating water in urinals and monitoring flow in real-time, which results in financial savings and a reduction in environmental impact by limiting water waste. The system was developed using presence sensors, a flow sensor, a microcontroller (*NodeMCU ESP32*), a control relay, and a solenoid valve. The hardware was programmed using the *Arduino* development platform. This hardware operates in conjunction with a *PostgreSQL* database and a *Node.js* web server. The system's main advantage lies in the range of technologies used, encompassing everything from embedded systems for data collection to a web platform with an interface for visualization, control, and data analysis.

Keywords: Sustainability; Monitoring; Control; Web Systems; Internet of Things;

LISTA DE FIGURAS

Figura 2.1 – Pinagem ESP32.	19
Figura 2.2 – IDE Arduino.	20
Figura 2.3 – Esquema de um Relé.	21
Figura 2.4 – Sensor PIR - Componente e Funcionamento.	22
Figura 2.5 – Sensor PIR - Módulo Comercial	22
Figura 2.6 – Esquema e imagem real de um solenoide.	23
Figura 2.7 – Osciloscópio mostrando pulsos de um sensor de fluxo com sensor Hall.	24
Figura 2.8 – Sensor de efeito Hall (Hidrômetro) - visão oblíqua.	25
Figura 2.9 – Módulo Regulador de Tensão Step Down Buck XL4015 5A.	26
Figura 2.10–Postgres - pgAdmin 4, tabela "livros".	29
Figura 2.11–Postgres - pgAdmin 4, tabela "emprestimo".	29
Figura 2.12–MongoDB - Compass - documento "emprestimos".	30
Figura 3.1 – Diagrama de conexão dos dispositivos.	33
Figura 3.2 – Projeto da placa PCB no Protheus.	34
Figura 3.3 – Visualização da impressão da placa PCB no Protheus.	34
Figura 3.4 – Unidade de controle e componentes.	34
Figura 3.5 – Diagrama de funcionamento.	35
Figura 3.6 – Código WifiManager.	36
Figura 3.7 – Tela de configuração de rede.	36
Figura 3.8 – pgAdmin 4 - Diagrama de Relacionamento de Entidade (ERD).	38
Figura 3.9 – Diagrama de Relacionamento de Entidade (ERD).	39
Figura 3.10–Diagrama servidor Node.Js.	40
Figura 3.11–Servidor Node.JS.	41
Figura 3.12–Tela de login.	41
Figura 3.13–Tela de cadastro.	42
Figura 3.14–Tela de monitoramento.	43
Figura 3.15–Tela de dados.	43
Figura 3.16–Tela de controle.	44
Figura 4.1 – Diagrama geral do protótipo.	45
Figura 4.2 – Protótipo do sistema.	46
Figura 4.3 – Fluxo sendo exibido em tempo real pelo software.	47
Figura 4.4 – Protótipo instalado.	49
Figura 4.5 – Protótipo instalado - monitoramento.	49
Figura 4.6 – Consumo sem atuação do protótipo.	50
Figura 4.7 – Consumo sem atuação do protótipo - Ampliado.	50
Figura 4.8 – Consumo com atuação do protótipo	51

Figura 4.9 – Consumo com atuação do protótipo - Ampliado.	51
Figura 4.10–Protótipo instalado.	53
Figura 4.11–Protótipo instalado - aberto.	54
Figura 4.12–Protótipo instalado - fechado.	55
Figura 4.13–Sensor de fluxo e solenoide instalados no encanamento do banheiro. . .	56
Figura 4.14–Consumo com atuação do protótipo.	56
Figura 4.15–Consumo com atuação do protótipo - Ampliado.	57
Figura 4.16–Consumo sem atuação do protótipo.	57
Figura 4.17–Consumo sem atuação do protótipo - Ampliado.	57

LISTA DE TABELAS

Tabela 2.1 – Especificações do ESP32.	18
Tabela 2.2 – Descrição dos botões da IDE Arduíno.	19
Tabela 3.1 – Tabela de dados técnicos, sensor de fluxo.	32
Tabela 3.2 – Tabela de dados técnicos, Válvula Solenoide Jin Sheng.	32
Tabela 4.1 – Resultados das amostragens.	48
Tabela 4.2 – Tabela de consumo mictório.	52
Tabela 4.3 – Tabela de consumo banheiro (CAMTUC).	58
Tabela 4.4 – Tabela de custos do protótipo do hardware.	59

LISTA DE SIGLAS

SGBD	Sistema Gerenciador de Banco de Dados.
IoT	Internet da Coisas (<i>Internet of Things</i>).
ULP	<i>Ultra Low Power</i> .
IDE	Ambiente Integrado de Desenvolvimento.
HTTP	<i>Hypertext Transfer Protocol</i> .
HALL	Fenômeno magnético descoberto em 1879 por Edwin Herbert Hall.

SUMÁRIO

1	INTRODUÇÃO	13
1.1	Escassez e Demanda por Água Potável	13
1.2	Projeções Futuras e Soluções Tecnológicas	13
1.3	Objetivos	14
1.3.1	Objetivo geral	14
1.3.2	Objetivos específicos	15
1.4	Justificativa do trabalho	15
1.5	Metodologia	15
1.6	Estrutura do trabalho	16
2	REVISÃO BIBLIOGRÁFICA	17
2.1	Sistemas embarcados	17
2.2	Monitoramento de Vazão	17
2.3	Hardware	17
2.3.1	Hardware do ESP32	17
2.3.2	IDE do Arduino	18
2.3.3	Relé	20
2.3.4	Sensor PIR	21
2.3.5	Válvula Solenoide	22
2.3.6	Sensor de Fluxo - Hidrômetro Efeito Hall	24
2.3.6.1	Formula geral	24
2.3.7	Módulo Regulador de Tensão	25
2.4	Software	26
2.4.1	NodeJS	27
2.4.2	Servidor	27
2.4.3	<i>Visual Studio Code</i>	28
2.4.4	Banco de dados	28
3	DESENVOLVIMENTO	31
3.1	Prototipação do Hardware	31
3.2	Prototipação do banco de dados	37
3.3	Prototipação do sistema para interface com o usuário	40
4	RESULTADOS	45
4.1	Produto final	45
4.2	Testes realizados	46

4.2.1	Precisão	47
4.2.2	Mictório	48
4.2.3	Banheiro masculino (UFPA Campus Tucuruí)	52
4.3	Custo do Projeto	58
5	CONCLUSÃO	60
5.1	Trabalhos Futuros	60
	REFERÊNCIAS	62
	APÊNDICE A – REPOSITÓRIOS NO GITHUB	65

1 INTRODUÇÃO

1.1 Escassez e Demanda por Água Potável

A escassez de água potável é uma preocupação global crescente, impulsionada por fatores como crescimento populacional, mudanças climáticas e poluição. De acordo com um relatório da Organização das Nações Unidas (ONU), aproximadamente 2 bilhões de pessoas não possuem acesso a serviços de água potável, representando 26% da população mundial. Além disso, o uso global de água tem aumentado cerca de 1% ao ano nas últimas quatro décadas; essa tendência deve continuar até 2050 (UNIDAS, 2023).

No Brasil, a preocupação com a escassez de água é evidente. Um estudo da *GlobeScan*, em parceria com a *Circle of Blue* e o *WWF*, revelou que 81% dos brasileiros estão muito preocupados com a falta de água potável, percentual superior à média global de 58%. Além disso, 40% dos brasileiros já foram diretamente impactados por períodos de seca (BRASIL, 2023).

1.2 Projeções Futuras e Soluções Tecnológicas

Projeções indicam que a demanda por água potável no Brasil pode crescer quase 80% nas próximas duas décadas devido a fatores econômicos, demográficos e climáticos. Estima-se que, até 2040, os brasileiros demandarão 14,3 bilhões de metros cúbicos de água, um aumento de 43,5% em relação a 2017. Considerando o desperdício e os impactos das mudanças climáticas, essa demanda pode atingir 17,6 bilhões de metros cúbicos, representando um incremento de 76% (CLIMAINFO, 2020).

Diante desse cenário, tecnologias como sensores inteligentes, inteligência artificial (IA) e Sistemas de Informação Geográfica (SIG) vêm sendo aplicadas na gestão de recursos hídricos para otimizar a distribuição e reduzir perdas. Sensores IoT, combinados com algoritmos de IA, permitem a detecção de vazamentos em tempo real, promovendo intervenções mais rápidas e eficientes (G1 Meio Ambiente, 2023).

Os desafios impostos pela escassez de água potável exigem soluções inovadoras e o uso inteligente de tecnologias. A implementação de sistemas de monitoramento e otimização no uso da água se torna essencial para garantir a sustentabilidade desse recurso vital para as futuras gerações. Sistemas embarcados, caracterizados pela integração de hardware e software em um único dispositivo, permitem a automação de processos e a coleta de dados periodicamente ou em tempo real, oferecendo uma abordagem precisa e eficiente para a gestão da água. Estudos anteriores, como o de Silva (SILVA; OLIVEIRA; MARTINEZ, 2020), demonstram a eficácia de sistemas de automação na redução do consumo de água em ambientes industriais e residenciais.

O presente trabalho propõe o desenvolvimento de um sistema de controle e monitoramento de vazão aplicado a um mictório, utilizando tecnologias de entrada, como o microcontrolador ESP32, sensor de vazão, sensor de proximidade e atuadores automáticos. O objetivo principal é criar um sistema capaz de medir, controlar e monitorar a vazão de água em tempo real, proporcionando uma significativa redução no desperdício e promovendo a economia de água. A implementação de sistemas de monitoramento de vazão tem demonstrado potencial para gerar economias significativas em instituições públicas e privadas. Por exemplo, um estudo da Faculdade de Ciências da Administração de Pernambuco (FCAP/UPE) revelou que o monitoramento contínuo do consumo de água resultou na redução do indicador médio de consumo de 12 L por agente consumidor por dia, no período de 2012 a 2014, para 9 L entre 2015 e 2017. Além disso, o índice médio mensal de vazamentos foi de 8%, indicando a importância da detecção e reparo rápidos para a conservação da água (SOARES; PRADO; SILVA, 2019).

Além disso, este sistema será integrado a uma plataforma web, permitindo o acesso remoto e o armazenamento de dados históricos para análises detalhadas. Esta abordagem visa não apenas oferecer uma solução técnica, mas também uma ferramenta prática e acessível. Segundo a pesquisa de Almeida (ALMEIDA; COSTA, 2018) destacou-se a importância da integração de sistemas de monitoramento com plataformas web para melhorar a acessibilidade e a eficiência do gerenciamento de recursos.

A motivação para este trabalho está na necessidade de adotar práticas mais sustentáveis e tecnológicas para a gestão da água, especialmente no contexto atual, em que a escassez de recursos se torna cada vez mais evidente. Ao desenvolver um sistema embarcado de controle e monitoramento de vazão, busca-se oferecer uma solução que alie inovação tecnológica, eficiência econômica e responsabilidade ambiental.

Nos capítulos seguintes, são detalhadas as etapas de desenvolvimento do sistema, incluindo a escolha dos componentes, a arquitetura do sistema, a implementação do software, os testes realizados e os resultados obtidos. Espera-se que este trabalho contribua significativamente para a literatura técnica e sirva como base para futuras inovações na área de gestão de recursos hídricos.

1.3 Objetivos

1.3.1 Objetivo geral

Desenvolver um sistema de controle e monitoramento de vazão em mictórios utilizando tecnologias de sistemas embarcados, com foco na redução do consumo de água e na eficiência operacional, promovendo economia e sustentabilidade em ambientes públicos ou privados.

1.3.2 Objetivos específicos

- Projetar e implementar a parte física do sistema:
 - Selecionar e configurar os sensores, atuadores e dispositivo, para medição precisa da vazão e controle de água.
 - Utilizar o microcontrolador ESP32 para gerenciar as leituras dos sensores e controlar os atuadores.
- Desenvolver o software de controle:
 - Programar o ESP32 para realizar a leitura dos sensores de vazão e processar os dados.
 - Implementar algoritmos de controle para gerenciar a vazão de água com base nas leituras dos sensores.
- Integrar o sistema com uma plataforma web:
 - Desenvolver um serviço web com interface para visualização dos dados de consumo de água em tempo real.
 - Implementar funcionalidades para armazenar históricos de consumo e gerar relatórios.
- Realizar testes e validação do sistema:
 - Testar o sistema em condições controladas para verificar a precisão das medições e a eficácia do controle automatizado.
 - Validar a integração com a plataforma web e assegurar que os dados são transmitidos de forma segura e precisa.
 - Analisar a redução do consumo de água e os benefícios econômicos decorrentes da implementação do sistema.

1.4 Justificativa do trabalho

A gestão de recursos hídricos é uma preocupação crescente no mundo, especialmente em regiões onde este recurso não é abundante. A utilização inadequada e o desperdício de água representam desafios significativos para a sustentabilidade ambiental e econômica. Em ambientes de utilização pública, como mictórios, é comum o desperdício causado pela má utilização dos usuários. A falta de monitoramento pode resultar em perdas financeiras e impactos ambientais. Diante desse cenário, a implementação de um sistema de controle e monitoramento da vazão de água se apresenta como uma solução tecnológica viável e necessária.

1.5 Metodologia

A metodologia utilizada para o desenvolvimento deste trabalho deu-se em quatro etapas principais. Inicialmente, foi realizada uma revisão bibliográfica sobre todas as técnicas e equipamentos disponíveis para a realização do projeto.

A partir desse estudo, foram utilizadas as seguintes tecnologias:

- Plataforma de desenvolvimento do Arduino em conjunto com linguagem C/C++ (ARDUINO, 2022);
- Microcontrolador ESP32 (ESPRESSIF, 2022);
- Sensor de fluxo com efeito Hall (WIKIPEDIA, 2024a), sensor PIR (RAYZEEK, 2025), e atuadores, como relé (CROMATEK, 2021) e válvula solenoide (WIKIPEDIA, 2024b);
- Editor de código-fonte Visual Studio Code (MICROSOFT, 2023), juntamente com o ambiente Node.js (NODEJS.ORG, 2024); e a linguagem de programação JavaScript (ECMA, 2024);
- Sistema gerenciador de banco de dados (SGDB) Postgres (POSTGRESQL, 2023).

1.6 Estrutura do trabalho

Este trabalho está estruturado em cinco capítulos que abordam os diversos aspectos do desenvolvimento e aplicação de um sistema de controle e monitoramento de vazão. A seguir, apresenta-se um resumo da estrutura geral do trabalho.

A introdução apresenta o contexto e a motivação para o desenvolvimento de um sistema de controle e monitoramento de vazão, enfatizando a importância da gestão eficiente dos recursos hídricos.

O capítulo 2 explora conceitos relacionados aos sistemas embarcados e aos sensores de vazão, além de revisar as tecnologias de comunicação sem fio aplicadas a sistemas de monitoramento. Também são descritas as especificações dos componentes de hardware e software, bem como a arquitetura do sistema.

No capítulo 3, é mostrado o desenvolvimento do sistema, com foco na implementação prática. São discutidos o *design* do *hardware*, a implementação do software, a integração do sistema de comunicação e os testes de funcionalidade e desempenho realizados para assegurar a operabilidade do sistema.

O capítulo 4 apresenta os resultados dos testes, custos e análises. São discutidos os dados coletados, comparando-os com os objetivos definidos e com sistemas existentes, avaliando a eficiência e a eficácia do sistema desenvolvido.

A conclusão resume os principais achados do trabalho, evidenciando as contribuições para a área de gestão de recursos hídricos. Também são discutidas as limitações do estudo e apresentadas sugestões para pesquisas futuras.

2 REVISÃO BIBLIOGRÁFICA

2.1 Sistemas embarcados

Sistemas embarcados são computadores projetados para realizar tarefas específicas. Geralmente fazem parte de um dispositivo maior e são projetados para funcionar continuamente sem intervenção humana. Os sistemas embarcados são projetados para executar uma função específica ou um conjunto de funções, o que não é o caso de computadores de uso geral, como PCs ou laptops. Wolf (WOLF, 2012) afirma que eles estão presentes em uma variedade de dispositivos, incluindo eletrodomésticos, automóveis, equipamentos médicos e equipamentos de telecomunicações.

Os sistemas embarcados geralmente possuem um microcontrolador ou microprocessador, memória, interfaces de entrada e saída e software especial. O hardware é um pouco diferente dependendo da aplicação, então geralmente é otimizado para sua função. O software é desenvolvido de forma a garantir que o sistema seja confiável e esteja em conformidade com os requisitos da aplicação. Os sistemas embarcados são um recurso comum em muitos setores porque são baratos e fáceis de instalar. Entre as aplicações notáveis estão automação industrial, automobilística, médica e internet das coisas (IoT).

2.2 Monitoramento de Vazão

O monitoramento do fluxo é essencial para a gestão de recursos hídricos, pois permite a medição precisa da vazão de água em um sistema. Sensores de efeito Hall, amplamente utilizados em sistemas embarcados, possibilitam a detecção e a medição do fluxo de água em tempo real. Os dados coletados por esses sensores são fundamentais para otimizar o uso da água, regular atividades industriais e detectar vazamentos. Um exemplo disso é a Faculdade de Ciências da Administração de Pernambuco (FCAP/UPE), que reduziu o consumo médio de água de 12 L por agente consumidor por dia (2012-2014) para 9 L (2015-2017) por meio do monitoramento contínuo e da identificação de vazamentos (SOARES; PRADO; SILVA, 2019).

2.3 Hardware

2.3.1 Hardware do ESP32

O ESP32 é um dispositivo que dispõe de microcontroladores de baixo custo e consumo energético associados a um microprocessador e módulos de comunicação *WIFI* e *Bluetooth* integrados. O dispositivo desenvolvido pela *Espressif Systems* (empresa chinesa com sede em Xangai) e fabricado pela TSMC, possui como microprocessador o Tensilica Xtensa LX6, existente em duas versões, single-core e dual-core. O modelo dual-core (dois

núcleos) é o mais utilizado, e ainda conta com um coprocessador secundário ULP (*Ultra Low Power coprocessor*) de 8 MHz com baixo consumo, 150 μ A. O ULP é programado na interface de desenvolvimento da própria desenvolvedora do dispositivo, a ESP-IDF, esse ambiente integrado de desenvolvimento (IDE) utiliza a linguagem C++, isso possibilita um maior controle de todos os componentes e funcionalidades que o dispositivo entrega, porém diminui a abstração da programação característica da *Arduino IDE* e suas bibliotecas de código aberto (*open-source*) que facilitam o desenvolvimento dos projetos. Além disso, a placa de desenvolvimento possui algumas proteções e comodidades, como um regulador de tensão que permite alimentar o dispositivo com até 5 *volts*, botões para *reset* e desbloqueio de *bootloader* para o carregamento de novo *firmware*.

Tabela 2.1 – Especificações do ESP32.

Especificações	ESP32
MCU	Xtensa® LX6 32-bit
Wi-Fi 802.11 b/g/n	HT40
Bluetooth	Bluetooth 4.2 LE
Frequência	240 MHz
SRAM	520 kB
SPI flash	4 MB

Fonte: Adaptado de (ESPRESSIF, 2020).

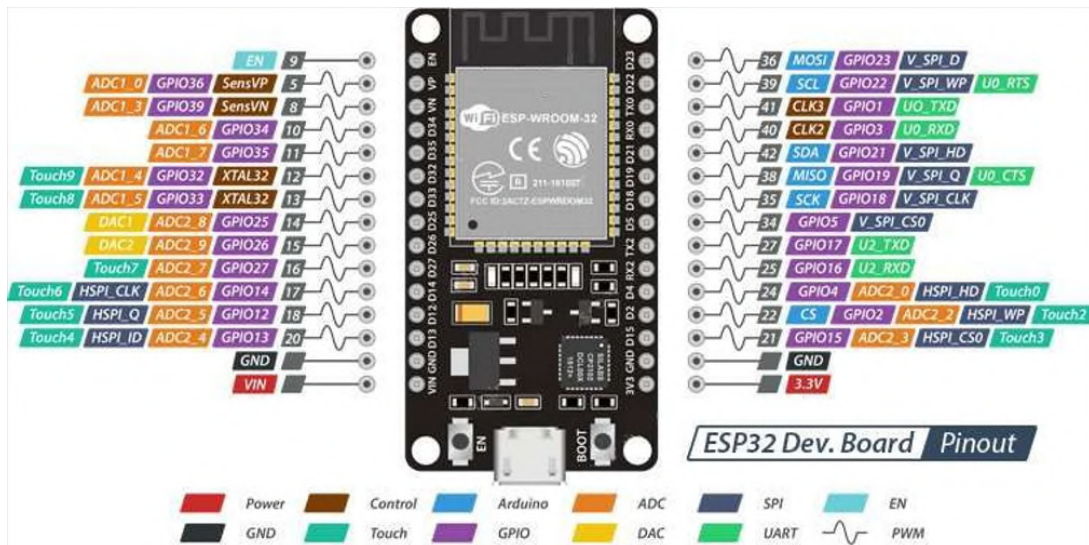
(ALBUQUERQUE, 2020) realiza uma breve explicação sobre os pinos GPIO (*General Purpose Input Output*) que são portas que servem como interface entre o microcontrolador e periféricos como sensores, atuadores ou até outros circuitos integrados e microcontroladores. A imagem mostrada na Figura 2.1 apresenta a pinagem de um DOIT Esp32 DevKit v1 de 30 pinos. Ele tem 25 pinos GPIOs com níveis de tensão 0 V e 3,3 V (não tolera 5 V) para níveis digitais 0 e 1 respectivamente, com corrente máxima de 12 mA por GPIO a.

2.3.2 IDE do Arduino

O ambiente de desenvolvimento consiste em um software gratuito, onde será escrita a sequência de instruções que serão interpretadas pelo Arduino. Ele se conecta ao hardware para realizar a comunicação e carregar o código desenvolvido. Os códigos escritos neste ambiente de desenvolvimento são chamados de *Sketches*, que são salvos com a extensão “.INO”. Quando se inicia o Ambiente para Desenvolvimento Integrado (IDE) encontra-se a área para escrever o software, a barra de ferramentas, o console de textos, que exibe uma lista completa de erros no código e o resultado das instruções enviadas ao Arduino, e os seguintes botões descritos na tabela a seguir na tabela 2.2.

Há também alguns comandos adicionais oferecidos para facilitar o desenvolvimento, como por exemplo: *Copy for fórum*, encontrado dentro do menu editar, que torna possível copiar o código para postar em fóruns, *Copy as HTML*, oferece opção de copiar como

Figura 2.1 – Pinagem ESP32.



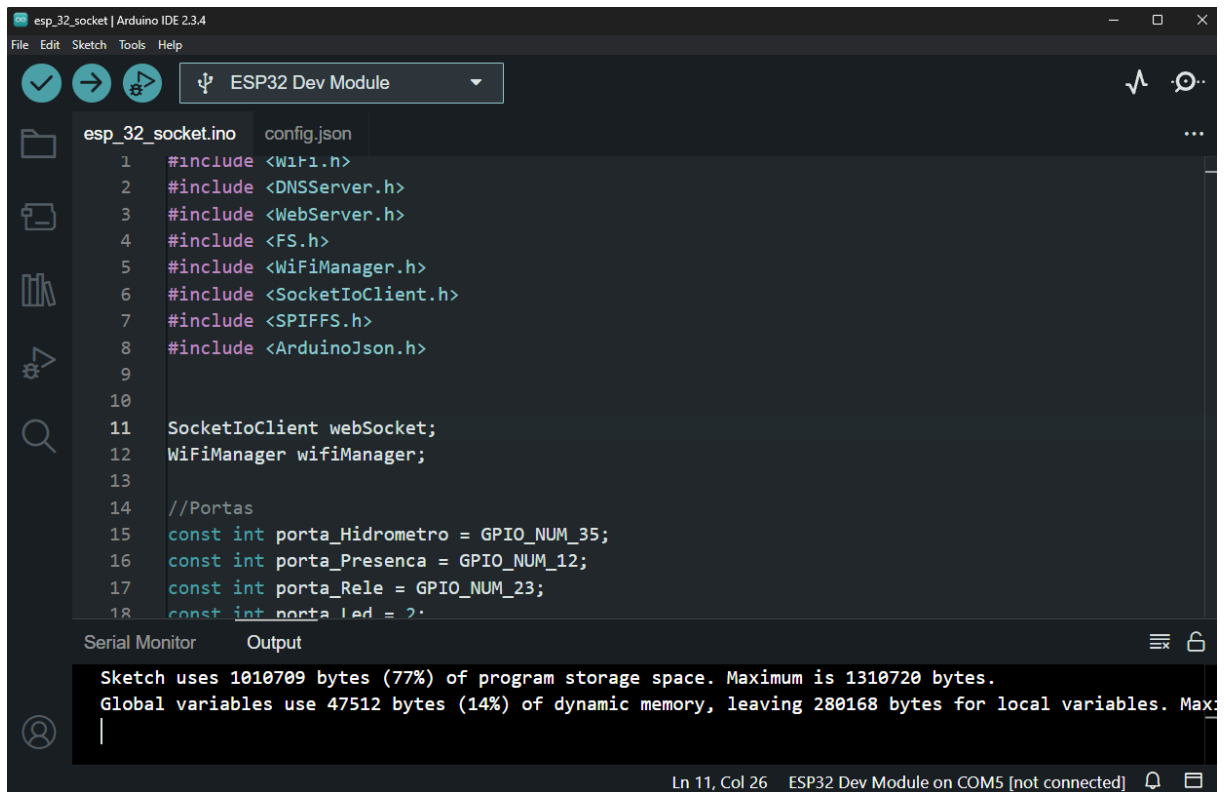
Fonte: (ALBUQUERQUE, 2020)

Tabela 2.2 – Descrição dos botões da IDE Arduino.

<i>Verify:</i>	Verifica erros no código.
<i>Upload:</i>	Compila o código e carrega para a placa.
<i>New:</i>	Cria um novo esboço.
<i>Open:</i>	Apresenta um menu com vários códigos prontos.
<i>Save:</i>	Salva o <i>sketch</i> atual.

HTML para inserir em páginas Web, *Import Library*, que adiciona uma biblioteca ao projeto, *Examples*, onde encontra-se diversos exemplos de código prontos, entre outras opções que tornam essa ferramenta muito simples de trabalhar. A IDE do Arduino foi escolhida como ambiente de programação do ESP32 pois possui um vasto acervo de exemplos, estes exemplos, em sua maioria, podem ser aplicados ao ESP32 ou a outros microcontroladores fora da família Arduino. A IDE do Arduino é mostrada na Figura 2.2, o *Sketch* contido na imagem da IDE é o código de exemplo *Blink*:

Figura 2.2 – IDE Arduino.

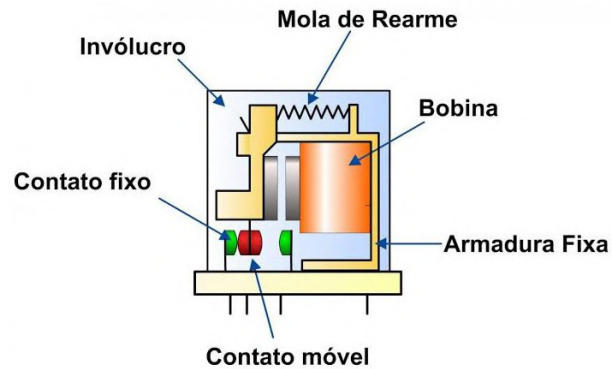


Fonte: *Autor*

2.3.3 Relé

Relé é um componente eletrônico que tem capacidade de ligar ou desligar um sinal a partir de outro, funcionando como um interruptor e com muitas aplicações. Segundo Eletronics (2019), o relé é composto por uma bobina e um contato preso a uma mola, que conecta nos terminais nas posições de Normalmente Aberto (NA) e Normalmente Fechado (NF). Na Figura 2.3 (CROMATEK, 2021), apresenta-se um esquemático de um relé; quando a bobina é energizada, cria-se um campo magnético que desloca o contato, fazendo-o passar da posição fechada para a posição aberta.

Figura 2.3 – Esquema de um Relé.



Fonte: Cromatek, 2021.

De acordo com Carneiro (2018), existem alguns tipos de relés, eles são: relés de tipo batente, utilizados quando uma pequena quantidade de energia na bobina pode ligar e desligar circuitos com altas tensões e correntes. Relé térmico, usado em aplicações em que uma elevação de temperatura deve ser sentida para ativar, entre outros modelos.

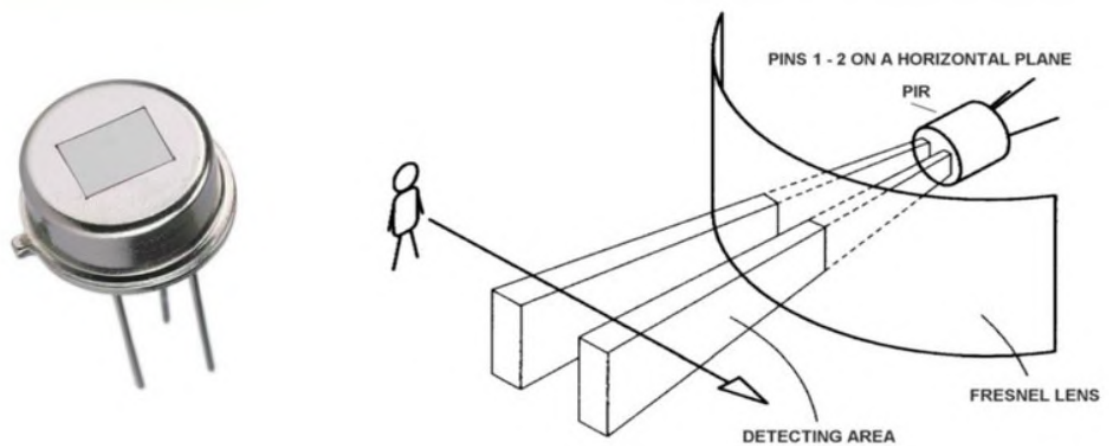
2.3.4 Sensor PIR

O Sensor PIR (*Passive Infrared Sensor*) é um dispositivo utilizado para detectar a presença de objetos que emitem radiação infravermelha, como seres humanos e animais. Segundo Eletronics (2020), ele opera captando variações na radiação infravermelha do ambiente e acionando um sinal elétrico quando há movimento dentro de sua área de detecção.

Esse sensor possui um regulador de alcance, que permite ajustar a distância de detecção, que geralmente varia de alguns centímetros a vários metros. Além disso, conta com um temporizador de saída, responsável por definir o tempo em que o sinal de ativação permanece acionado após a detecção, tornando-o adaptável a diferentes aplicações.

A Figura 2.4 (SILÍCIO, 2020) ilustra um típico sensor PIR, com sua estrutura e funcionamento básico. Já a Figura 2.5 apresenta um módulo comercial do sensor PIR, com imagens da vista frontal (Figura 2.5a) e da vista traseira (Figura 2.5b).

Figura 2.4 – Sensor PIR - Componente e Funcionamento.

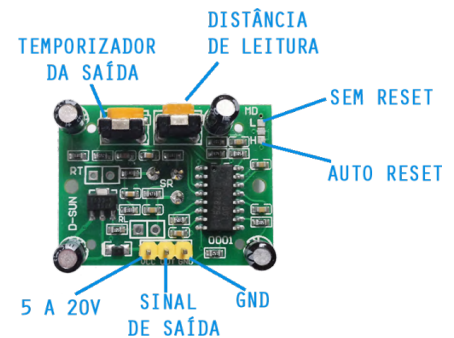


Fonte: *Vida de Silício*, 2020.

Figura 2.5 – Sensor PIR - Módulo Comercial



(a) Vista frontal.



(b) Vista traseira.

Fonte: (ROBOCORE, 2024)

De acordo com (OLIVEIRA, 2019), sensores PIR são amplamente utilizados em sistemas de alarme, iluminação automática e aplicações que exigem detecção de movimento sem contato físico.

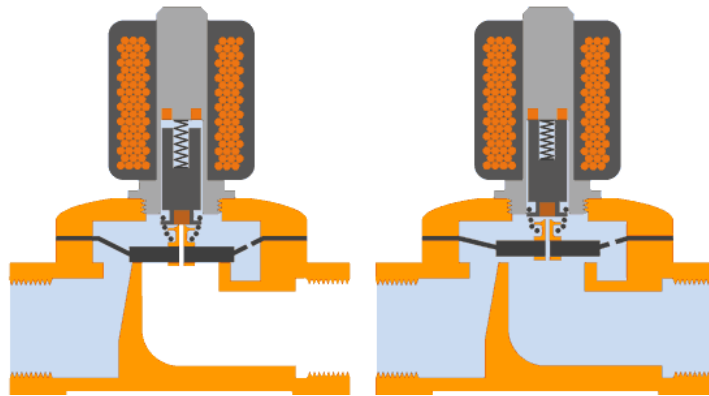
2.3.5 Válvula Solenoide

Segundo (SILVEIRA, 2017), a válvula solenoide é uma peça eletromecânica usada para controlar o fluxo de líquidos e gases. Seu funcionamento comum é baseado em uma bobina elétrica com um êmbolo ferromagnético no centro, que fecha um orifício em uma membrana de borracha, quando está em repouso. Quando uma corrente elétrica passa pela bobina, um campo magnético é criado, o que faz o êmbolo se mover e abrir o orifício, o que equaliza a pressão e possibilita o levantamento da membrana, que antes era pressionada pelo fluido e permanecia fechada.

Essa tecnologia é amplamente empregada em diversos setores industriais ao redor do mundo, desde sistemas de aquecimento e ar comprimido até automação industrial e sistemas de irrigação. Sua versatilidade faz com que as aplicações para a válvula solenoide sejam praticamente ilimitadas, sendo uma peça fundamental em muitos equipamentos e processos industriais.

Existem vários tipos de válvulas solenoide, dentre elas estão as com operação direta, indireta e semi-direta. Tendo também como opção o estado NA ou NF. Além das válvulas eletro-hidráulicas, onde não existe bobina e sim um pistão hidráulico. Esta, por sua vez, é empregada mais comumente no campo industrial, devido à sua robustez e possibilidade de controlar precisamente a quantidade de abertura, que, por consequência, influi na quantidade de vazão. Na Figura 2.6 é possível observar o esquema (Figura 2.6a) e a imagem (Figura 2.6b) (CITISYSTEMS, 2019) de uma válvula indireta e normalmente fechada, modelo comumente utilizado.

Figura 2.6 – Esquema e imagem real de um solenoide.



(a) Figura de uma válvula solenoide.

Fonte: *CitiSystems, (2017)*



(b) Exemplar de válvula solenoide.

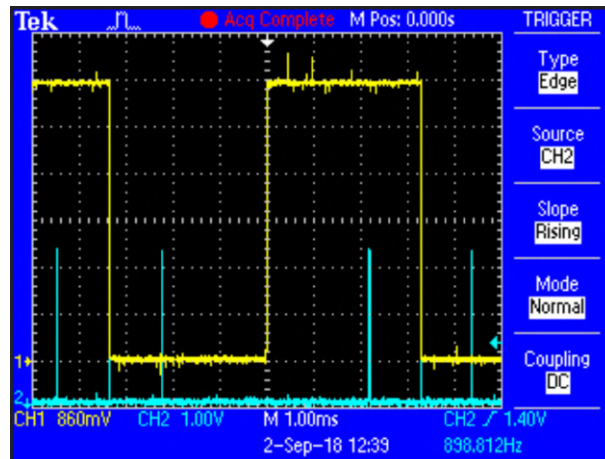
Fonte: *Autor*

2.3.6 Sensor de Fluxo - Hidrômetro Efeito Hall

O sensor de fluxo de efeito Hall é um dispositivo utilizado para medir a taxa de fluxo de líquidos ou gases em diversos sistemas. Esse tipo de sensor utiliza o princípio do efeito Hall para detectar e quantificar o fluxo. Seus principais componentes são o rotor ou turbina, os ímãs e o próprio sensor de efeito Hall.

O funcionamento ocorre por meio de um rotor ou turbina que gira com a passagem do fluido. Os ímãs fixados no rotor geram um campo magnético, que é detectado pelo sensor de efeito Hall. Com a rotação, são gerados pulsos elétricos proporcionais à velocidade do rotor, conhecidos como Modulação por Largura de Pulso (PWM, do inglês *Pulse Width Modulation*) Figura 2.7 . Esses pulsos correspondem à taxa de fluxo do fluido (WEBSTER; EREN, 2017).

Figura 2.7 – Osciloscópio mostrando pulsos de um sensor de fluxo com sensor Hall.



Fonte: (DUMBO, 2018)

2.3.6.1 Formula geral

A vazão do fluido pode ser determinada a partir dos pulsos do sensor Hall, usando a seguinte fórmula:

A equação para a vazão instantânea Q é dada por:

$$Q = P \times V_p \quad (\text{EQ. 1})$$

onde:

- Q é a vazão instantânea em mililitros por segundo (mL/s);
- P é o número de pulsos registrados em um intervalo de 1 segundo;
- V_p é o volume de fluido por pulso, determinado pelo sensor e pelo diâmetro da tubulação (exemplo: 2,25 mL/pulso).

Para obter a vazão em litros por minuto (L/min), converte-se da seguinte forma:

$$Q_{L/\min} = \frac{Q \times 60}{1000} \quad (\text{EQ. 2})$$

Substituindo Q :

$$Q_{L/\min} = \frac{P \times V_p \times 60}{1000} \quad (\text{EQ. 3})$$

Assim, o consumo total de fluido em um intervalo de tempo T (em segundos) é:

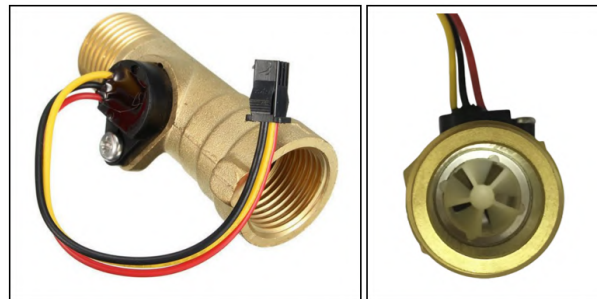
$$C = \sum_{i=1}^T P_i \times V_p \quad (\text{EQ. 4})$$

Ou, em litros:

$$C_L = \sum_{i=1}^T \frac{P_i \times V_p}{1000} \quad (\text{EQ. 5})$$

Este tipo de sensor é bastante preciso e durável, com aplicações em diversas indústrias, como automotiva, para medição de fluxo de combustível, sistemas de irrigação e indústrias químicas. A capacidade de medir diferentes tipos de fluidos com precisão o torna um instrumento versátil e confiável.

Figura 2.8 – Sensor de efeito Hall (Hidrômetro) - visão oblíqua.



Fonte: *Autor*

2.3.7 Módulo Regulador de Tensão

Os reguladores de tensão são dispositivos essenciais em circuitos eletrônicos, garantindo que a tensão fornecida a um circuito seja estável e adequada para seu funcionamento (GIOVANNI.DE.CASTRO, 2024). Um dos modelos mais utilizados para aplicações de corrente contínua é o regulador de tensão step down buck XL4015.

Este módulo é baseado no chip XL4015, capaz de converter tensões de entrada de 4V a 38V para uma tensão de saída ajustável entre 1,25V e 36V. Ele opera com eficiência elevada e pode fornecer até 5A de corrente de saída, sendo ideal para aplicações que exigem alta potência e estabilidade.

A Figura 2.9 apresenta uma imagem do módulo XL4015.



Figura 2.9 – Módulo Regulador de Tensão Step Down Buck XL4015 5A.

O regulador XL4015 é amplamente utilizado em projetos de fontes de alimentação ajustáveis, carregadores de bateria e sistemas embarcados. Sua capacidade de dissipação térmica é aprimorada pelo dissipador de calor integrado, garantindo operação estável mesmo sob altas correntes.

2.4 Software

Segundo Pressman (2006), "sob o ponto de vista da Engenharia de Software, um software é um conjunto composto por instruções de computador, estruturas de dados e documentos. As instruções (ou código) produzem os resultados esperados, de acordo com os requisitos definidos."(PRESSMAN, 2006).

Ainda para (PRESSMAN, 2006), o software tornou-se uma peça central nas atividades cotidianas da sociedade contemporânea, por desempenhar papel importante na transformação digital, por possuir função polivalente na sociedade, acabou permeando em diferentes setores. Na esfera econômica, sistemas de gestão empresarial otimizam processos, ampliando a eficiência e a competitividade. No âmbito educacional, softwares inovadores oferecem métodos de aprendizado avançados. Em áreas críticas, como saúde, softwares de diagnóstico e gestão de pacientes elevam a qualidade dos serviços.

A integração de softwares com dados do mundo real encontra uma ilustração clara na Internet das Coisas (IoT). Dispositivos interconectados coletam e compartilham dados em tempo real, influenciando diversas áreas. Um exemplo prático é a gestão inteligente de

tráfego em áreas urbanas, onde sensores em semáforos ajustam os tempos de sinalização conforme o fluxo de veículos, otimizando o tráfego.

2.4.1 NodeJS

O Nodejs é um ambiente de execução JavaScript assíncrono orientado a eventos, inicialmente projetado para aplicações web escaláveis, com alta taxa de fluxo e baixa latência. Várias conexões podem ser efetuadas ao mesmo tempo, devido a assincronicidade das funções e pelo fato de quase nenhuma efetuar operações de E/S diretamente, não existe Deadlock. Isso o torna uma ótima opção para servir como base para uma biblioteca web ou para um framework, como por exemplo, Express (NODEJS.ORG, 2024; EXPRESSJS.COM, 2024). Seu event-loop é exposto como parte do ambiente de execução e não como uma biblioteca. Em outros sistemas há sempre uma chamada bloqueante para inicializar o event-loop como por exemplo `EventMachine::run()`. Essa execução seria inicializada em thread principal e poderia sofrer interferências de outros processos em execução pelo sistema operacional (S.O), como concorrência de recursos compartilhados pelo S.O.

2.4.2 Servidor

Na Engenharia da Computação, os servidores desempenham um papel fundamental, agindo como entidades que fornecem recursos e serviços em uma rede. Um servidor, em sua essência, é um sistema dedicado a atender solicitações de outros computadores, chamados clientes.

Ao explorar os primórdios da comunicação digital, nos deparamos com a ARPANET (*Advanced Research Projects Agency Network*, em português, Rede da Agência de Pesquisas em Projetos Avançados), responsável pela criação da primeira rede de computadores. Desenvolvido nos anos 1960 pelo Departamento de Defesa dos EUA como um meio seguro para transmitir dados militares sigilosos e para interligar os departamentos de pesquisa por todo o país. O projeto culminou no estabelecimento dos fundamentos para a comunicação entre diferentes tipos de computadores. A ARPANET foi o precursor do TCP/IP, conjunto de protocolos que é a espinha dorsal da Internet moderna. (ABBATE, 1994).

O protocolo HTTP (*Hypertext Transfer Protocol*) é um exemplo clássico. Ele gerencia a transferência de dados na *World Wide Web*, permitindo que os navegadores solicitem e recebam páginas da web. Quando um usuário acessa um site, seu navegador envia uma solicitação ao servidor HTTP, que responde enviando os dados necessários para exibir a página.

Outro componente vital na comunicação entre computadores é o servidor baseado em sockets. Sockets são pontos de extremidade para a comunicação em uma rede e permitem que os programas troquem dados. Um servidor de socket aguarda ativamente

por solicitações, estabelece uma conexão quando solicitado e facilita a transferência de informações entre os dispositivos conectados.

Em resumo, os servidores são elementos-chave que viabilizam a comunicação e o compartilhamento de recursos. Seja na entrega de páginas web através de servidores HTTP ou na troca dinâmica de dados via *sockets*, essas estruturas desempenham um papel essencial na arquitetura de redes de computadores, moldando a infraestrutura que sustenta nossa conectividade digital.

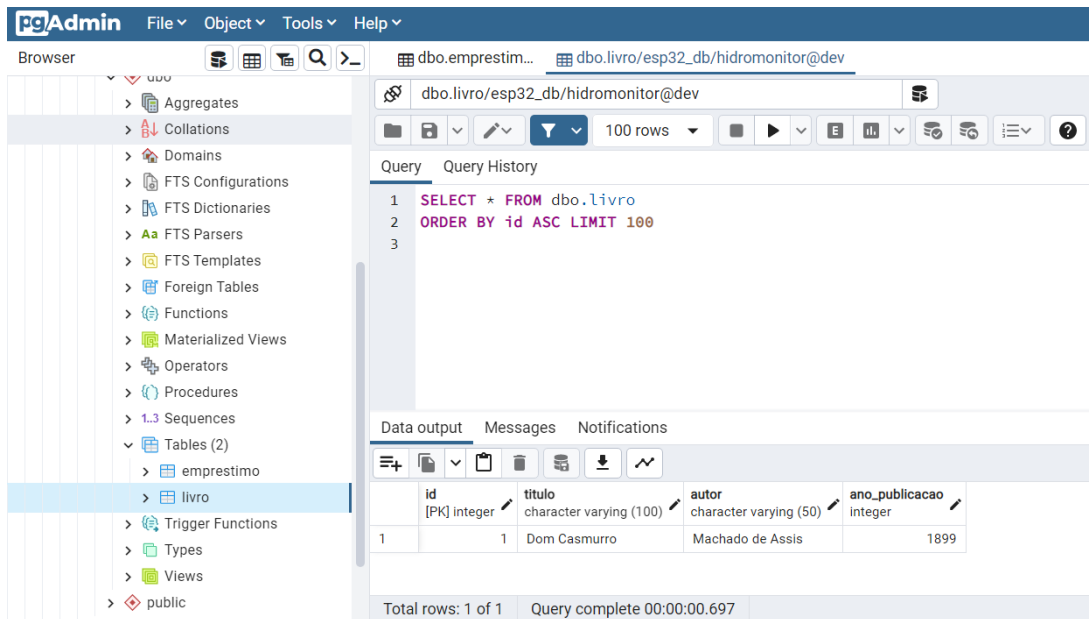
2.4.3 *Visual Studio Code*

O *Visual Studio Code* ou VSCode, como popularmente é conhecido, é um editor de código-fonte poderoso e leve. Foi lançado pela Microsoft no ano de 2015, durante um evento da marca. Possui versões para os principais sistemas operacionais para Windows, macOS e Linux. Com ele é possível programar em diversas linguagens como JavaScript, TypeScript, C++, C#, Java, Python, PHP, Go, entre outras. Apesar de pertencer a uma das gigantes da tecnologia, a Microsoft, o VSCode é gratuito e *Open Source*. Isso significa que a comunidade de desenvolvedores que utiliza essa ferramenta pode realizar contribuições para aprimoramento e adicionar recursos, que, em alguns casos, se tornam oficiais.

2.4.4 Banco de dados

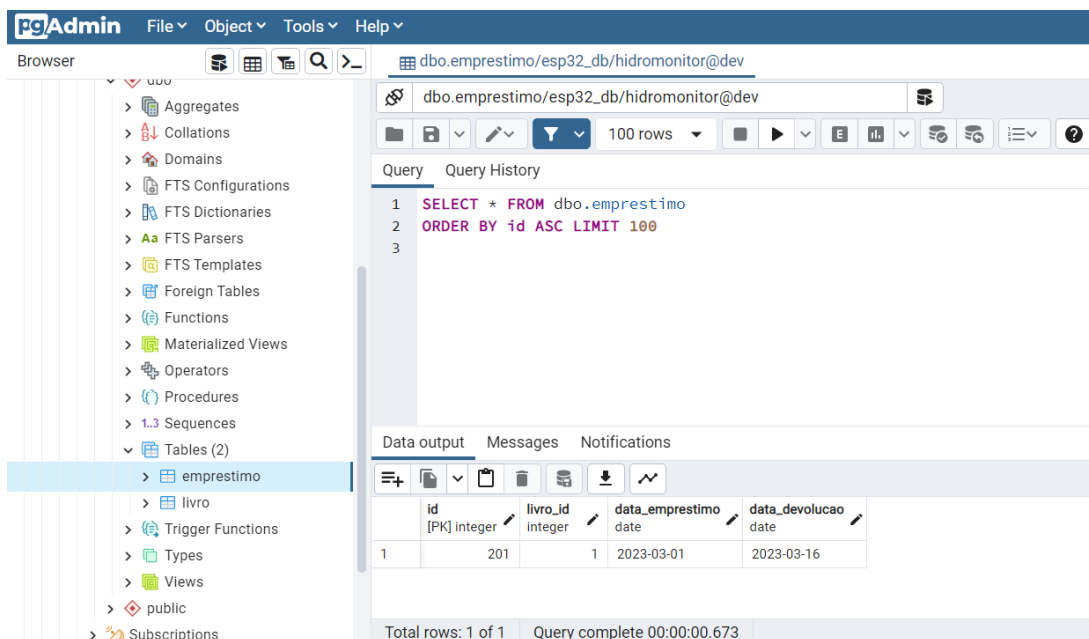
O banco de dados é uma ferramenta fundamental no universo da tecnologia da informação, sendo uma estrutura organizada para armazenar, gerenciar e recuperar dados de maneira eficiente. Ele desempenha um papel central em diversas aplicações, desde sistemas empresariais até plataformas *web*. Nesse contexto, os dados são estruturados e armazenados de modo a facilitar sua manipulação e recuperação. O *PostgreSQL* é um exemplo de banco de dados relacional, no qual os dados são organizados em tabelas, com relações entre elas. Como se observa no exemplo a seguir, a interface do software de administração de banco de dados, *pgAdmin 4*, nas Figuras 2.10 e 2.11.

Figura 2.10 – Postgres - pgAdmin 4, tabela "livros".



Fonte: Autor

Figura 2.11 – Postgres - pgAdmin 4, tabela "emprestimo".

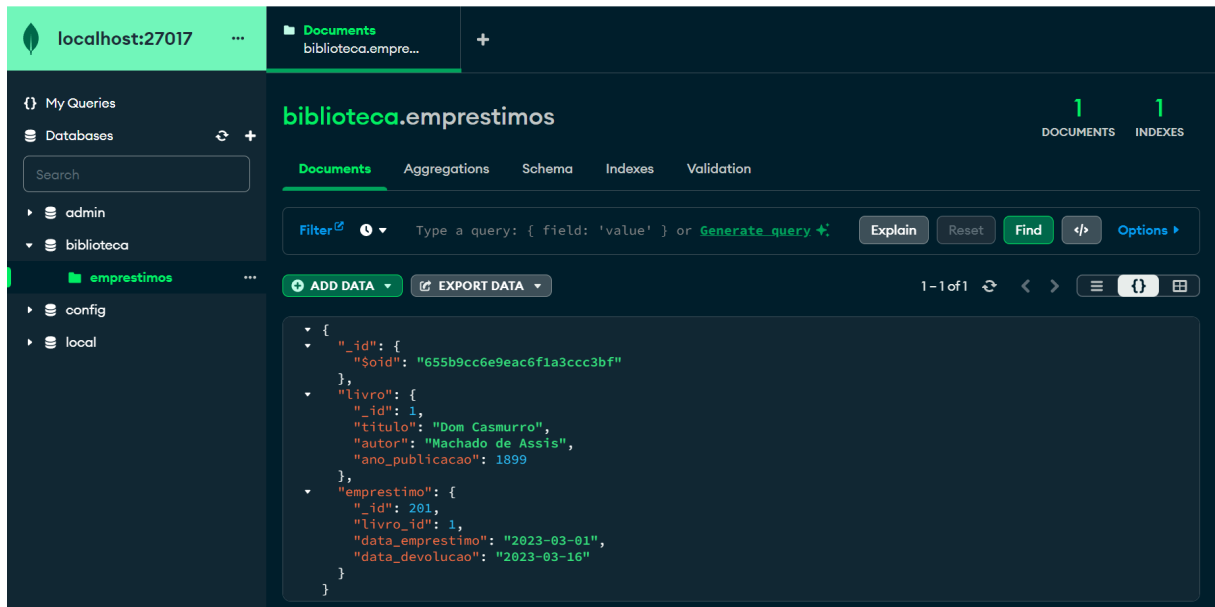


Fonte: Autor

Já em ambientes NoSQL (a sigla significa 'Not Only SQL', ou 'Não Somente SQL', que é a Linguagem de Consulta Estruturada) (COODESH, 2020), como o MongoDB, que é um banco de dados orientado a documentos, desenvolvido para armazenar grandes volumes de dados de forma flexível e escalável (MONGODB, 2020), a abordagem é diferente. Utiliza-se documentos no formato *JSON*. Cada documento é equivalente a uma linha em

uma tabela relacional, mas sem a necessidade de um esquema fixo. A Figura 2.12 mostra um exemplo de um arquivo JSON através da ferramenta de gerenciamento *MongoDB Compass*.

Figura 2.12 – MongoDB - Compass - documento "emprestimos".



Fonte: *Autor*

Em resumo, enquanto bancos de dados relacionais, como o PostgreSQL, seguem uma estrutura tabular, os bancos de dados NoSQL, como o MongoDB, adotam abordagens mais flexíveis, que permite esquemas dinâmicos e adaptáveis às necessidades específicas da aplicação. Essa diversidade de modelos oferece opções valiosas aos desenvolvedores na escolha da tecnologia mais adequada para cada cenário. Foi escolhido utilizar um banco de dados SQL devido à sua capacidade de lidar com dados estruturados e inter-relacionados, garantindo integridade referencial. A necessidade de transações ACID (Atomicidade, Consistência, Isolamento, Durabilidade) e consistência de dados, essenciais em sistemas críticos, foi outro fator decisivo. Além disso, a robustez das consultas complexas e a maturidade das soluções SQL asseguram estabilidade e escalabilidade vertical. Por fim, a ampla gama de ferramentas e o ecossistema bem desenvolvido oferecem suporte contínuo ao longo do ciclo de vida do sistema.

3 DESENVOLVIMENTO

Neste capítulo, são abordados todos os passos, ferramentas e equipamentos utilizados para implementar o projeto proposto. Uma descrição geral da construção e desenvolvimento do hardware, banco de dados e software.

3.1 Prototipação do Hardware

Neste projeto foi proposta a implementação de um sistema de acionamento automático para mictórios utilizando ESP32 em conjunto com sensores e atuadores. O modelo conta com uma placa ESP32 interligada com um sensor de proximidade que, ao detectar a presença de uma pessoa em frente ao mictório, enviará o sinal para o microcontrolador que, posteriormente, acionará uma válvula solenoide, responsável pela liberação de água. Por meio de um sensor de fluxo, também será possível monitorar o consumo de água utilizado em todo o processo.

O ponto inicial para realizar a construção do protótipo físico foi o levantamento dos componentes necessários, através de seus *datasheets* e arquivos de referência. A partir deste filtro, foi realizada a escolha dos seguintes equipamentos: ESP32 para processamento de dados e comunicação, sensor de fluxo para mensurar vazão, uso de um sensor PIR para detecção da presença de indivíduos, um relé, que em conjunto com uma válvula solenoide acionará a abertura do fluxo de água e uma fonte genérica de 12 *volts* 2 *amperes* para alimentar os componentes eletrônicos. De posse do microcontrolador com recursos integrados, tais quais portas SPI, *Bluetooth*, *Wi-Fi*, a construção iniciou.

Na mensuração do fluxo de vazão, foi utilizado o sensor de fluxo de efeito Hall, acoplado ao microcontrolador. Com o sensor devidamente alimentado com 5 *volts*, o sinal PWM gerado pelo terceiro condutor restante foi conectado à porta 22 do ESP32, permitindo aferir a velocidade instantânea da hélice interna quando rotacionada.

Conforme o manual do sensor da Tabela 3.1, o parâmetro estabelecido pelo fabricante é de 2,25 ml a cada revolução. Portanto, para calcular a vazão em litros por minuto, utilizou-se a seguinte fórmula: $\mathbf{P} * 2,25 * 60 / 1000$, sendo \mathbf{P} a quantidade de pulsos aferidos em um segundo, que corresponde exatamente ao tempo de interrupção definido para a porta 22 do microcontrolador.

Tabela 3.1 – Tabela de dados técnicos, sensor de fluxo.

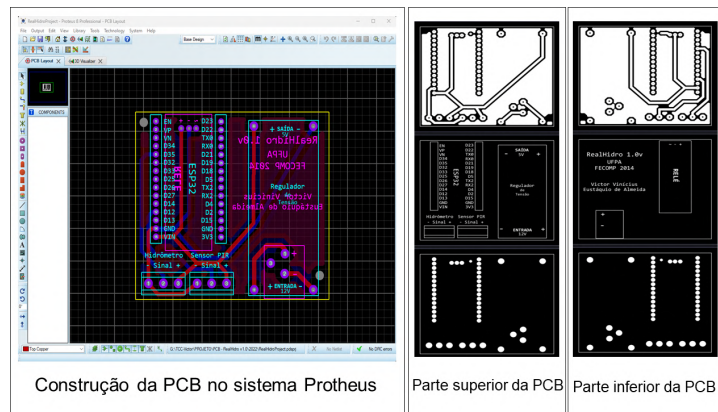
Característica	Descrição
Modelo	YF-B8
Material	Cobre
Vazão	1 a 30 L/m $\pm 3\%$
Alimentação	3,5 24V
Pulso (máx)	4,5V
Fluxo de Pulso	$F=2,25*Q \pm 5\%$ (2,25 ml/pulso)
Pressão (max)	1,75 Mpa
Temperatura (max)	100° C

Para permitir e ou bloquear o fluxo no sistema, foi adicionada uma válvula solenoide naturalmente fechada 3.2, que, quando acionada, alterna seu regime para aberta, visando o funcionamento correto. Esta válvula foi acoplada a um relé, tendo a função de receber o sinal do microcontrolador e liberar a alimentação da válvula, assim alternando o seu estado para aberta por um tempo determinado. O acionamento do sistema se dará à detecção de presença de um usuário no ambiente. O sensor PIR identifica a presença e a distância mínima para o acionamento. O sensor foi conectado à porta 15 no microcontrolador. Através dessas ligações, o ESP32 pode controlar e receber todas as informações do sistema. Nos anexos desse trabalho seguem os repositórios com os códigos utilizados para a recepção de dados e controle de cada equipamento. Na figura 3.1, estão ilustradas todas as conexões físicas presentes no projeto.

Tabela 3.2 – Tabela de dados técnicos, Válvula Solenoide Jin Sheng.

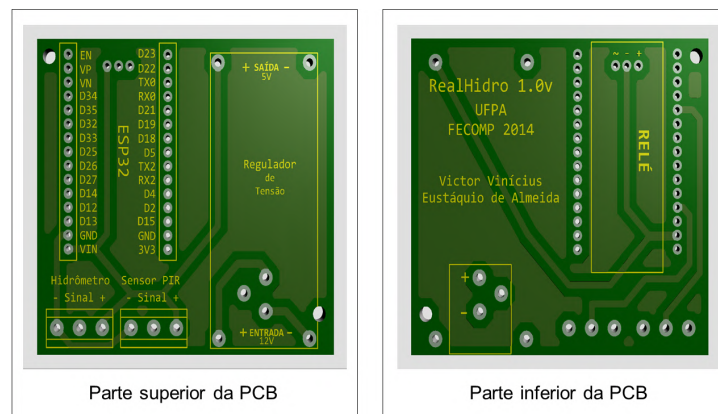
Característica	Descrição
Modelo	TS101
Regime	Normalmente Fechado
Material	Plástico (PA)
Vazão (max)	16,7 L/m
Alimentação	12V 0,6A (DC)
Pressão (min)	0,02 Mpa
Pressão (max)	1,6 Mpa
Temperatura	-30 a 100° C

Figura 3.2 – Projeto da placa PCB no Protheus.



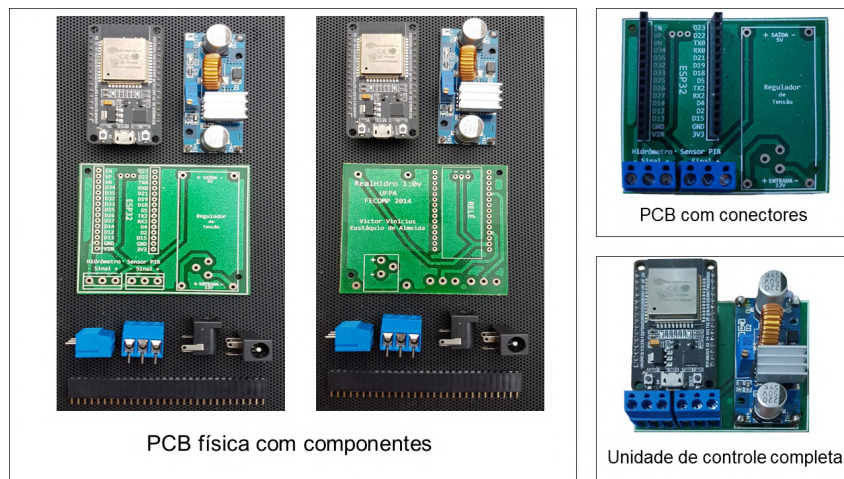
Fonte: Autor (software Protheus)

Figura 3.3 – Visualização da impressão da placa PCB no Protheus.



Fonte: Autor (software Protheus)

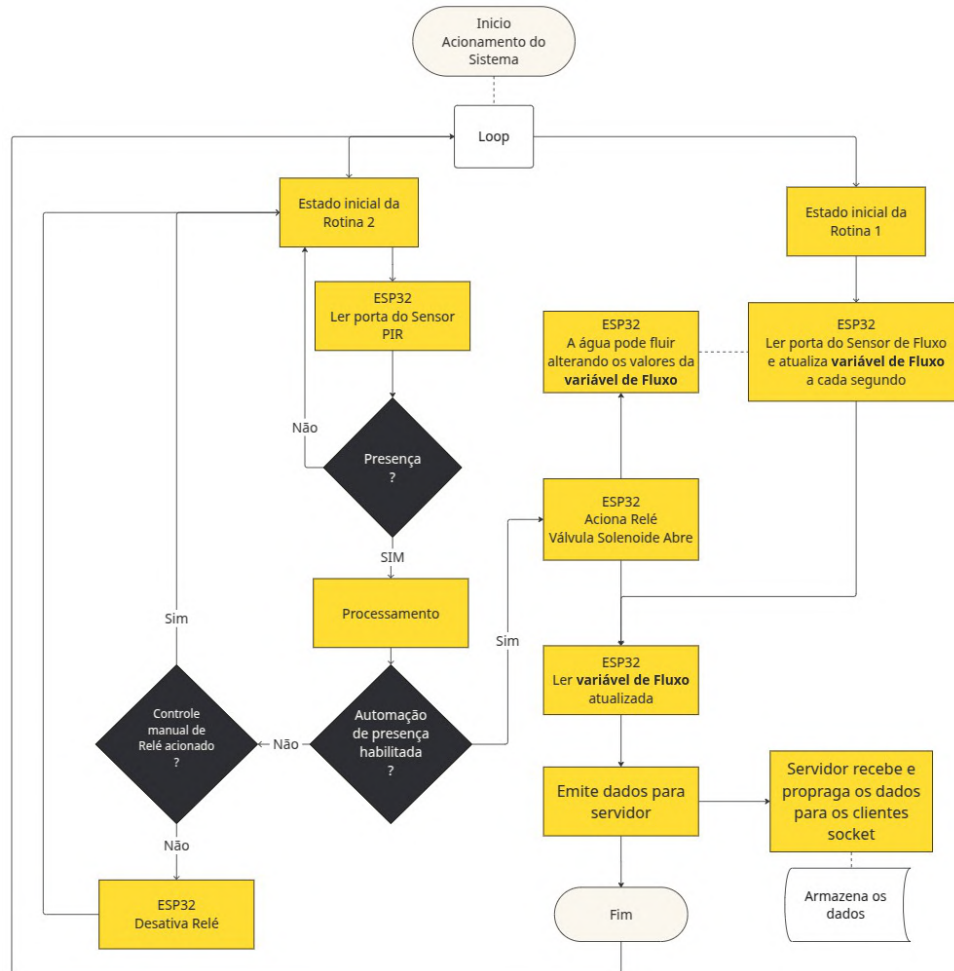
Figura 3.4 – Unidade de controle e componentes.



Fonte: Autor

Visando melhor descrição e entendimento do funcionamento do hardware, criou-se um diagrama de funcionamento do sistema, presente na figura 3.5.

Figura 3.5 – Diagrama de funcionamento.



Fonte: Autor

O ESP32 possui um Wi-Fi integrado em seu hardware, essa ferramenta foi utilizada para facilitar o envio de informações para o usuário. Para conectar o dispositivo à rede é necessário gravar as informações da rede em seu firmware, com o intuito de simplificar a configuração de novas redes no equipamento sem a necessidade de gravar um novo firmware, utilizamos a biblioteca WifiManager no código fonte, implementando uma sub-rotina antes da execução do código principal, permitindo adicionar novas redes com o equipamento em uso. Através dessas redes, o equipamento pode fazer o envio dos dados coletados, seguem as imagens do código figura 3.6 e tela de conexão das redes pelo WifiManager figura 3.7.

Figura 3.6 – Código WifiManager.

```

WiFiManager | Arduino 1.8.20 Hourly Build 2022/04/25 09:33
Arquivo Editar Sketch Ferramentas Ajuda

WiFiManager $
#include <WiFi.h> //ESP32 Core WiFi Library
#include <DNSServer.h> //Local DNS Server used for redirecting all requests to the configuration portal ( https://github.com/zhouhan0126/DNSServer---esp32 )
#include <WebServer.h> //Local WebServer used to serve the configuration portal ( https://github.com/zhouhan0126/WebServer-esp32 )
#include <WiFiManager.h> // WiFi Configuration Magic ( https://github.com/zhouhan0126/WIFIMANAGER-ESP32 ) >> https://github.com/tzapu/WiFiManager (ORIGINAL)

void setup() {
  Serial.begin(115200);
  WiFiManager wm;

  wm.setWiFiAutoReconnect(true); /* Auto-Reconexão */
  wm.setConnectTimeout(10); /* Tempo aguardando Auto-Reconexão (10 segundos) */
  wm.setConfigPortalTimeout(60*3); /* Tempo no qual o modo de configuração WiFi ficará disponível (3 minutos) */

  /*Iniciando modo de configuração WiFi*/
  if(!wm.autoConnect("ESP_CONFIG_WIFI", "12345678")) {
    Serial.println("Desconectado!");
    Serial.println("Iniciando modo de Configuração");
  }
  else {
    Serial.println("Conectado!");
  }
}

void statusConexao () {
  WiFiManager wm;
  if(wm.getWLStatusString() != "WL_CONNECTED"){
    Serial.println("Conexão perdida!");
    ESP.restart();
  }
  delay(2000);
}

void loop() {
  Serial.println("Loop :)");
  statusConexao();
}

```

Fonte: Autor

Figura 3.7 – Tela de configuração de rede.



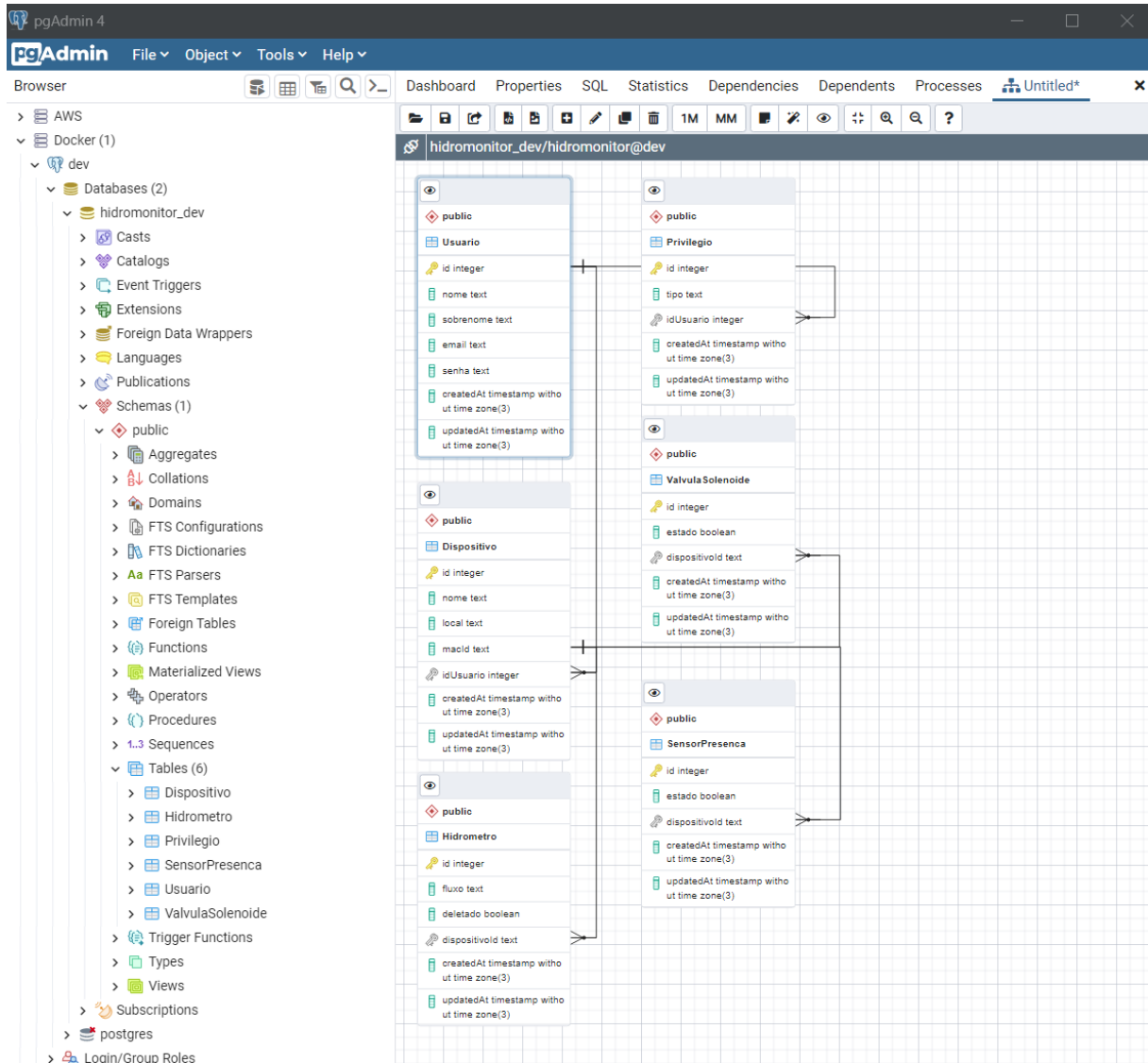
Fonte: Autor

3.2 Prototipação do banco de dados

Devido ao alto número de registros que o sistema pode gerar e à necessidade de armazenar os dados com perseverança e de forma segura, foi optado pela criação de um banco de dados SQL, ferramenta ideal para a tarefa. O banco implementado trata-se de um banco relacional PostgreSQL, a escolha da plataforma PostgreSQL deu-se a larga usabilidade para sistemas que sincronizam uma grande quantidade de dados, com alta taxa de leitura e escrita, similar ao sistema proposto. Durante a modelagem do banco, determinou-se que o banco irá armazenar os dados de volumetria hidráulica coletados pelo sensor de fluxo, usuários, dispositivos e permissões de acesso.

Após o levantamento de quais dados serão armazenados, foram definidas as tabelas que vão compor o banco, estas são "Usuario", "Privilegio", "Dispositivo", "ValvulaSolenoides", "SensorPresença". A Figura 3.8 mostra a interface do **pgAdmin 4** conectada ao banco de dados de desenvolvimento do sistema; ela apresenta todas as tabelas ao lado esquerdo e o diagrama de relacionamento de entidade ao lado direito.

Figura 3.8 – pgAdmin 4 - Diagrama de Relacionamento de Entidade (ERD).



Fonte: Autor (software pgAdmin4)

Logo abaixo segue a descrição das tabelas, campos e quais dados são armazenados:

”**Usuario**”: Essa tabela representa as pessoas que usam o sistema. Cada pessoa tem seu próprio perfil com nome, sobrenome, e-mail e senha. Elas também podem ter diferentes níveis de acesso, chamados de privilégios, e podem controlar vários dispositivos.

”**Privilegio**”: São as diferentes permissões que uma pessoa pode ter. Uma pessoa pode ter vários privilégios, como administrador ou usuário padrão.

”**Dispositivo**”: Armazena a relação de todos os microcontroladores que estão conectados ao nosso sistema. Cada microcontrolador recebe um nome, um nome para descrever a localização e possui número de identificação único **MAC**. Eles são atribuídos a um usuário específico e podem incluir dados adicionais, que estão presentes nas tabelas relacionadas, ”Hidrometro”, ”ValvulaSolenoide” e ”SensorPresenca”.

”**Hidrometro**”: Essa tabela é responsável por registrar o fluxo de água dos dispo-

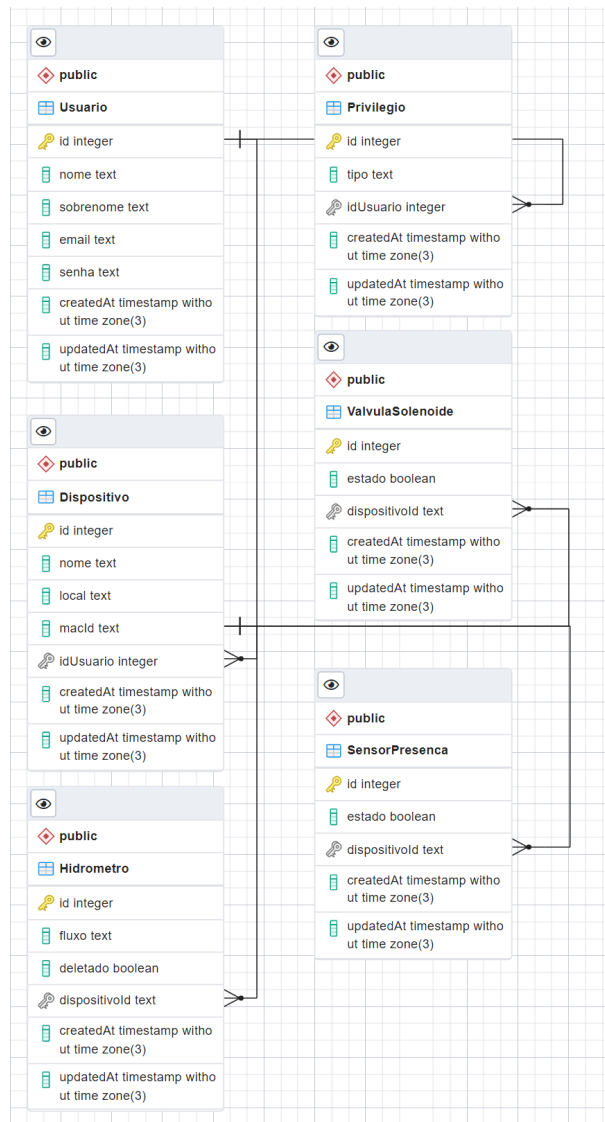
sitivos hidrômetros; cada hidrômetro está diretamente associado a um microcontrolador específico.

”**ValvulaSolenoide**”: São as válvulas que controlam o fluxo de água ou outro fluido. Elas podem estar abertas ou fechadas e estão vinculadas a um microcontrolador específico.

”**SensorPresenca**”: Essa tabela representa os sensores que detectam quando alguém está presente em uma determinada área. Os dados também estão vinculados a um microcontrolador específico.

Após todos esses processos, o banco de dados está completo, podendo ser utilizado para carregamento e descarregamento das informações. A figura 4.16 descreve o relacionamento entre as tabelas criadas dentro do banco de dados.

Figura 3.9 – Diagrama de Relacionamento de Entidade (ERD).



Fonte: Autor (software pgAdmin4)

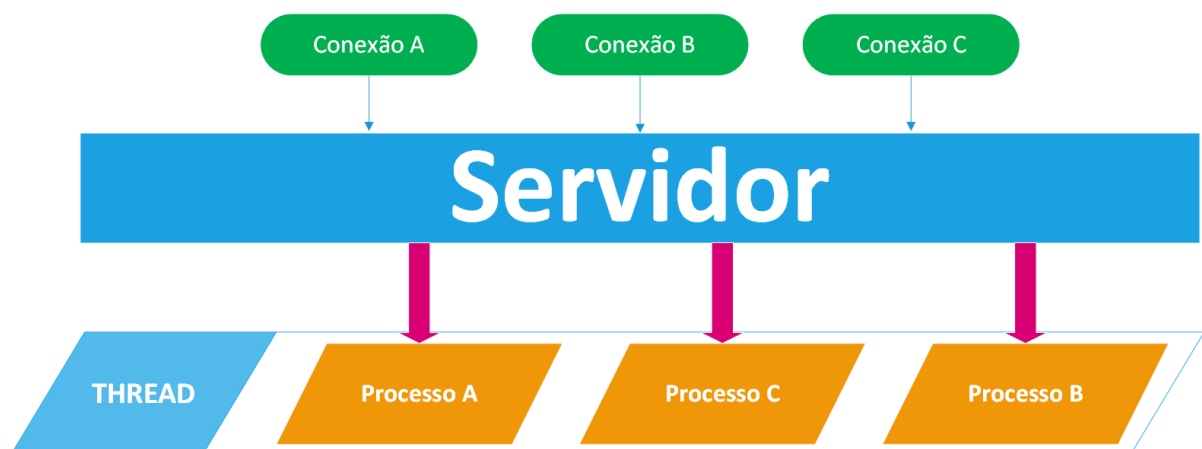
3.3 Prototipação do sistema para interface com o usuário

A visualização dos dados armazenados diretamente no banco de dados possui desvantagens, entre elas, a necessidade de conhecimento técnico para elaboração de consultas, limitação de acessos e abstração de dados por estarem sem tratamentos. Para melhorar a visualização dos dados obtidos, surgiu a necessidade de um sistema para fazer a interface homem-máquina, levando à escolha de um sistema *web*. A opção pelo sistema *web* se deu pela necessidade de implementação e instalação de um servidor *web* em apenas um dispositivo, que será responsável pela execução do sistema e poderá ser acessado em rede por qualquer equipamento com acesso à internet, sem necessidade de instalação.

A estrutura e o esquema de dados do banco criado possibilitam a implementação de uma aplicação para gerenciamento e monitoramento de dispositivos, como em sistemas de automação residencial ou industrial. Com isso, os usuários podem controlar, acompanhar e extrair relatórios de seus dispositivos de forma eficiente.

O servidor foi desenvolvido em NodeJs, utilizando a linguagem de programação Javascript, sua arquitetura consiste em um servidor que suporta multiconexões, executando as solicitações em threads (processamentos paralelos). O servidor possui conexão com o banco de dados citado anteriormente, podendo realizar consultas e inserções no banco, inclusive a validação de usuários. De acordo com as solicitações das conexões, ele enviará e receberá dados ou páginas, através dos métodos GET e POST. Estas páginas estão armazenadas em rotas dentro do servidor, as quais podem ser redirecionadas nas solicitações. A figura 3.10 mostra de forma resumida a estrutura de processamento utilizada internamente pelo Node.js e a figura 3.11 mostra os arquivos de códigos que compõem o projeto do servidor.

Figura 3.10 – Diagrama servidor Node.Js.



Fonte: Autor

Figura 3.11 – Servidor Node.JS.

```

44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

```

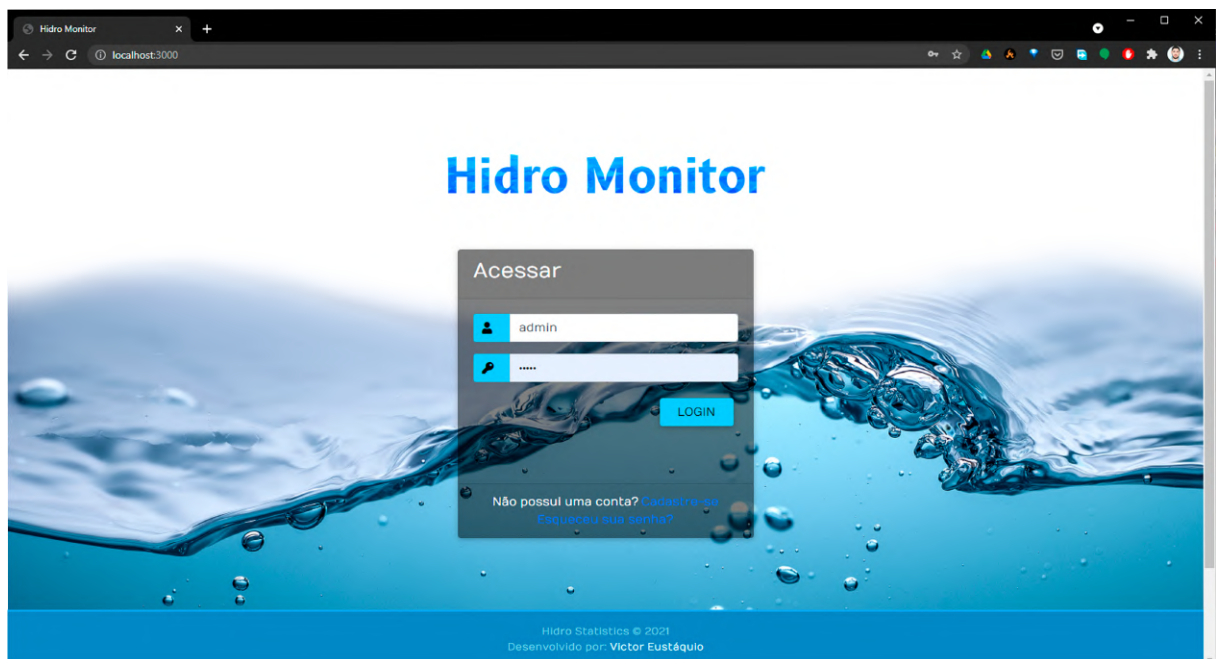
fluxo: '6.34',
  esplocal: 'Banheiro1'
}
EXECUTOU SQL => INSERT INTO 'esp32_db'.hidrometro_tbl ('espclient', 'local', 'fluxo') VALUES ('ESP32_Hidromedro1', 'Banheiro1', '6.34')
INSERTIDO!
{
  espclient: 'ESP32_Hidromedro1',
  fluxo: '5.94',
  esplocal: 'Banheiro1'
}
EXECUTOU SQL => INSERT INTO 'esp32_db'.hidrometro_tbl ('espclient', 'local', 'fluxo') VALUES ('ESP32_Hidromedro1', 'Banheiro1', '5.94')
INSERTIDO!
GET /charts 200 1.928 ms - 19
GET /varrele 304 2.088 ms - -

```

Fonte: Autor

Para a interface do usuário, foram implementadas quatro telas; estas telas servirão para autenticação, cadastros e visualização de dados ou equipamentos conectados ao sistema. A primeira tela do sistema é a tela de login, onde usuários já cadastrados possam efetuar seus logins no sistema.

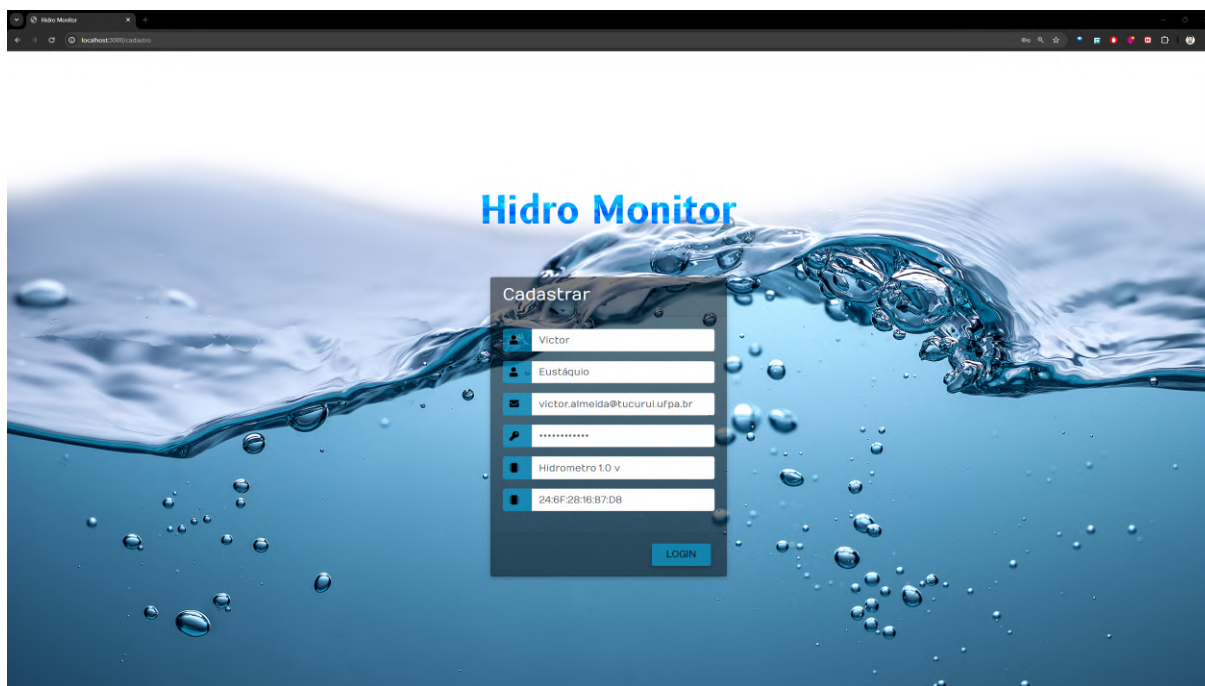
Figura 3.12 – Tela de login.



Fonte: Autor

A segunda tela, representada pela Figura 3.13, é a tela de cadastro de usuários e equipamentos; essa tela tem a função de cadastrar o usuário e o equipamento a ser monitorado. Seu formulário possui os campos NOME, SOBRENOME, E-MAIL, SENHA, HIDROMETRO, campo onde você pode definir um nome para o seu dispositivo e MAC para relacionarmos o dispositivo ao seu endereço físico.

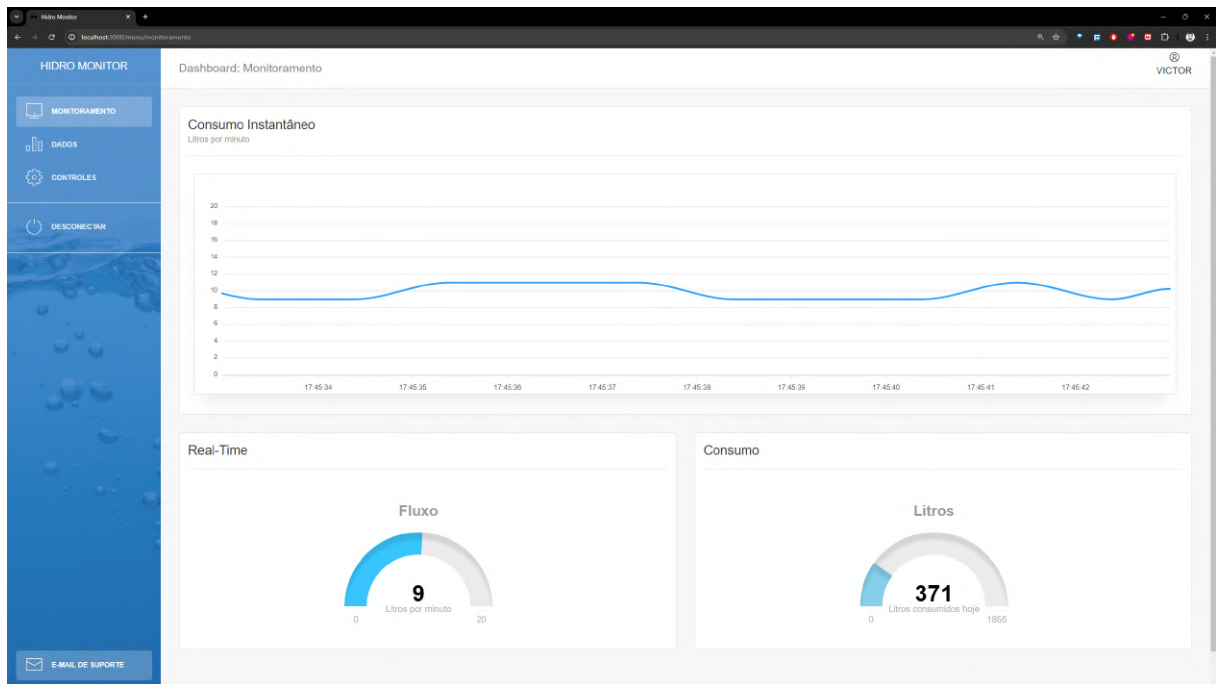
Figura 3.13 – Tela de cadastro.

A screenshot of a web browser displaying the registration page for 'Hidro Monitor'. The page features a blue background with a water splash effect. The title 'Hidro Monitor' is displayed in blue text. Below the title is a registration form titled 'Cadastrar' with the following fields: 'Nome' (filled with 'Victor'), 'Sobrenome' (filled with 'Eustáquio'), 'E-mail' (filled with 'victor.almeida@tucurui.ufpa.br'), 'Senha' (filled with '*****'), 'Hidrometro' (filled with 'Hidrometro 1.0 v'), and 'MAC' (filled with '24:6F:28:16:87:D8'). A 'LOGIN' button is located at the bottom right of the form.

Fonte: Autor

A tela inicial após o login é a tela de monitoramento, Figura 3.14; possui as informações de consumo dos equipamentos, como consumo instantâneo do equipamento em gráfico, consumo em tempo real e consumo do dia.

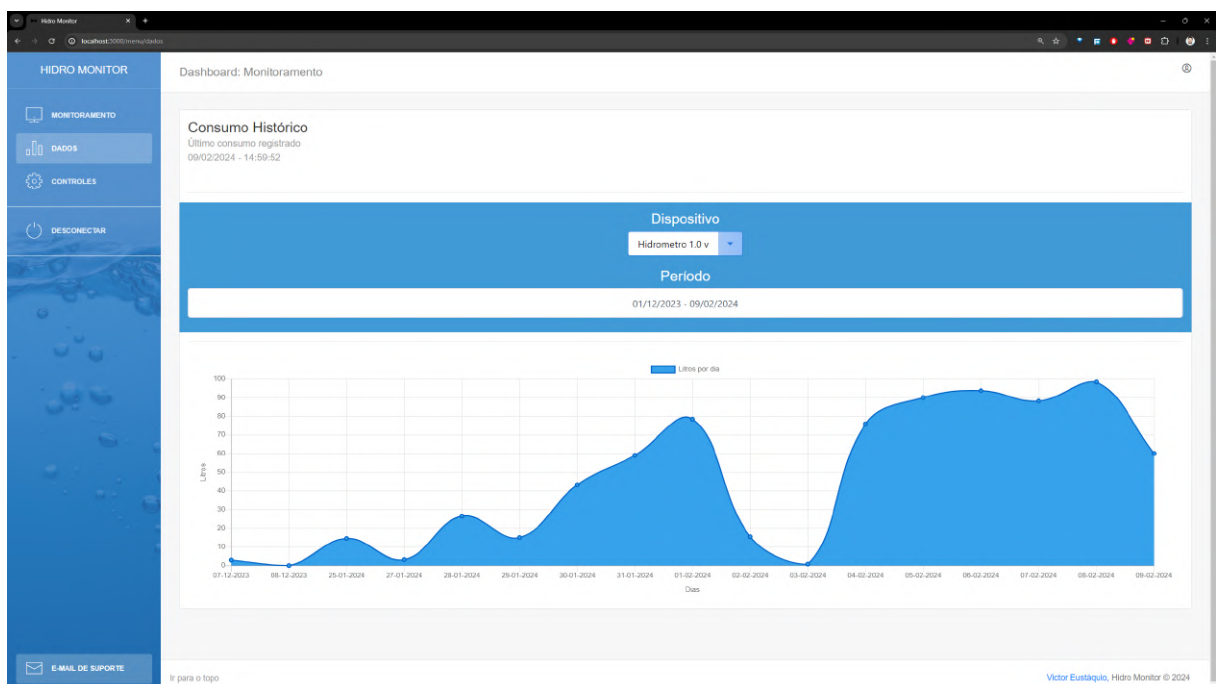
Figura 3.14 – Tela de monitoramento.



Fonte: Autor

Para realizar consultas de consumos entre dias, semanas ou mês, uma tela contendo filtros de datas foi criada, Figura 3.15, nela são apresentados os gráficos de consumo no período selecionado.

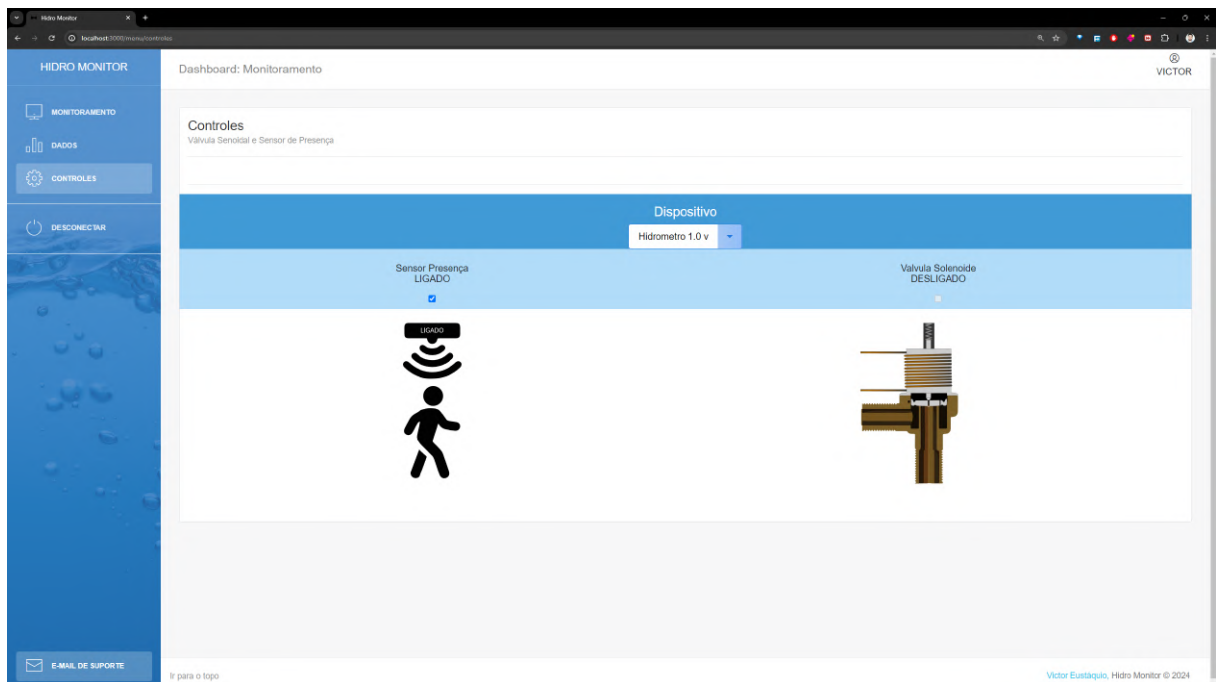
Figura 3.15 – Tela de dados.



Fonte: Autor

A tela de controle permite o acionamento remoto da vazão e a ativação do sensor de presença, permitindo manutenção ou interrupção do sistema. Nesta tela, o usuário pode ver todos os seus dispositivos, verificar o fluxo de água de seus hidrômetros, abrir ou fechar suas válvulas solenoides e verificar se os sensores de presença estão ativados. Na figura 3.16, mostra-se a tela de controle. Com o servidor *web* finalizado, o sistema pode renderizar a interface de login, monitoramento, controle e mostrar os dados emitidos pelo hardware em tempo real.

Figura 3.16 – Tela de controle.



Fonte: Autor

Além disso, esse esquema é flexível e pode ser expandido facilmente para incluir novos tipos de dispositivos ou funcionalidades no futuro.

Em resumo, o diagrama de relacionamento entre dados nos dá uma estrutura sólida para construir um aplicativo que permite aos usuários gerenciar e monitorar uma variedade de dispositivos de forma conveniente e eficiente.

4 RESULTADOS

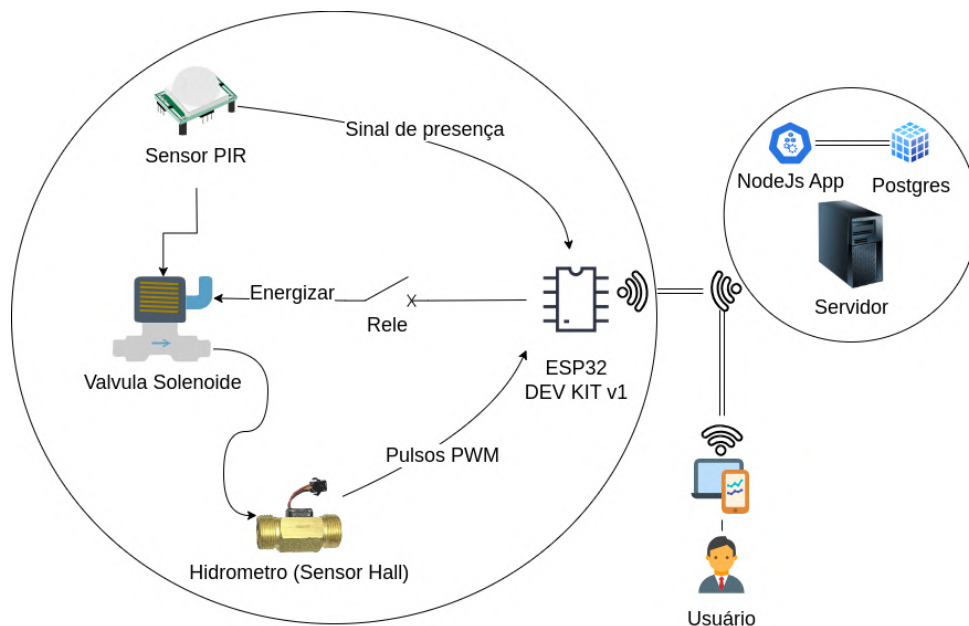
4.1 Produto final

O resultado mais significativo deste trabalho é a produção de um equipamento autônomo, criado para minimizar o consumo excessivo de água em banheiros, sejam públicos ou privados. O sistema não apenas reduz o consumo de água, mas também consegue monitorar em tempo real, armazenar e exibir dados de consumo, podendo ser uma ferramenta funcional para análises e ser configurado para diversas necessidades e aplicações.

O sistema como um todo pode ser considerado complexo devido à sua estrutura, composta por um hardware que faz o acionamento automático da água no mictório, monitorando e registrando, em paralelo, todo o consumo hídrico; um banco de dados responsável pelo armazenamento das informações geradas tanto pelo hardware quanto pelo software; e, por último, o principal diferencial: o servidor web, que funciona como uma interface para o usuário. Através dele, os dados podem ser visualizados, filtrados e cadastrados, e até mesmo comandos podem ser enviados ao hardware.

Para melhorar a visualização do sistema, foi construído um diagrama descritivo, apresentado nas figuras 4.1, e 4.2 .

Figura 4.1 – Diagrama geral do protótipo.



Fonte: Autor

Figura 4.2 – Protótipo do sistema.



Fonte: Autor

4.2 Testes realizados

Com todo o protótipo do sistema já implementado, o hardware e o software foram submetidos a pequenos testes de funcionalidade. Estes testes foram fundamentais para a obtenção de um equipamento atendendo à proposta inicial do trabalho. A parte física foi submetida a testes de acionamento da válvula solenoide, teste de aferição na vazão registrada pelo sensor e processada pelo hardware. Para testes na válvula solenoide, foram realizados acionamentos sucessivos; como resultado, foi obtido dados de todos acionamentos de abertura e fechamento do registro, sem falhas.

Foram realizados testes de cadastro no software, os quais foram validados com sucesso no banco de dados. Também foram feitos testes de navegação, nos quais o sistema web se mostrou rápido, responsivo e fluido ao alternar entre as telas. Para validar a comunicação entre o hardware e o software, acionou-se o fluxo de água no sistema, e a vazão foi exibida em tempo real na tela. A Figura 4.3 mostra o fluxo, em litros por minuto, coletado pelo sensor de fluxo.

Figura 4.3 – Fluxo sendo exibido em tempo real pelo software.



Fonte: Autor

4.2.1 Precisão

Para a avaliação do sensor de fluxo, um teste mais complexo foi aplicado. O teste consistiu em liberar fluxo de água pelo sensor de vazão e captar o volume de água em um recipiente com marcações, posteriormente comparando se a vazão aferida é igual ao volume captado pelo recipiente. A imagem 4.3 mostra como o teste foi realizado e a tabela 4.1 apresenta os resultados obtidos nos testes.

A precisão do sistema pode ser avaliada utilizando o erro percentual absoluto médio (MAPE - Mean Absolute Percentage Error):

Tabela 4.1 – Resultados das amostragens.

Amostragem	Volume Captado (L)	Precisão (%)	Fluxo Médio (L/m)
1	20.19	100.95	8.58
2	19.78	98.88	5.90
3	20.74	103.70	8.24

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{V_i - V_{ref}}{V_{ref}} \right| \times 100\% \quad (\text{EQ. 1})$$

Onde: - V_i é o volume medido na i -ésima amostragem, - V_{ref} é o volume de referência (20L), - n é o número total de amostras.

Substituindo os valores obtidos:

$$MAPE = \frac{1}{3} \left(\left| \frac{20.19 - 20}{20} \right| + \left| \frac{19.78 - 20}{20} \right| + \left| \frac{20.74 - 20}{20} \right| \right) \times 100\% \quad (\text{EQ. 2})$$

$$MAPE = \frac{1}{3} (0.95 + 1.12 + 3.70) \times 100\% \quad (\text{EQ. 3})$$

$$MAPE \approx 1.92\% \quad (\text{EQ. 4})$$

O resultado indica um erro percentual absoluto médio de aproximadamente 1.92%, o que sugere que, apesar das interferências como a falta de pressurização do sistema e a variação da pressão, o sensor de fluxo apresenta uma boa precisão na medição do volume de água.

4.2.2 Mictório

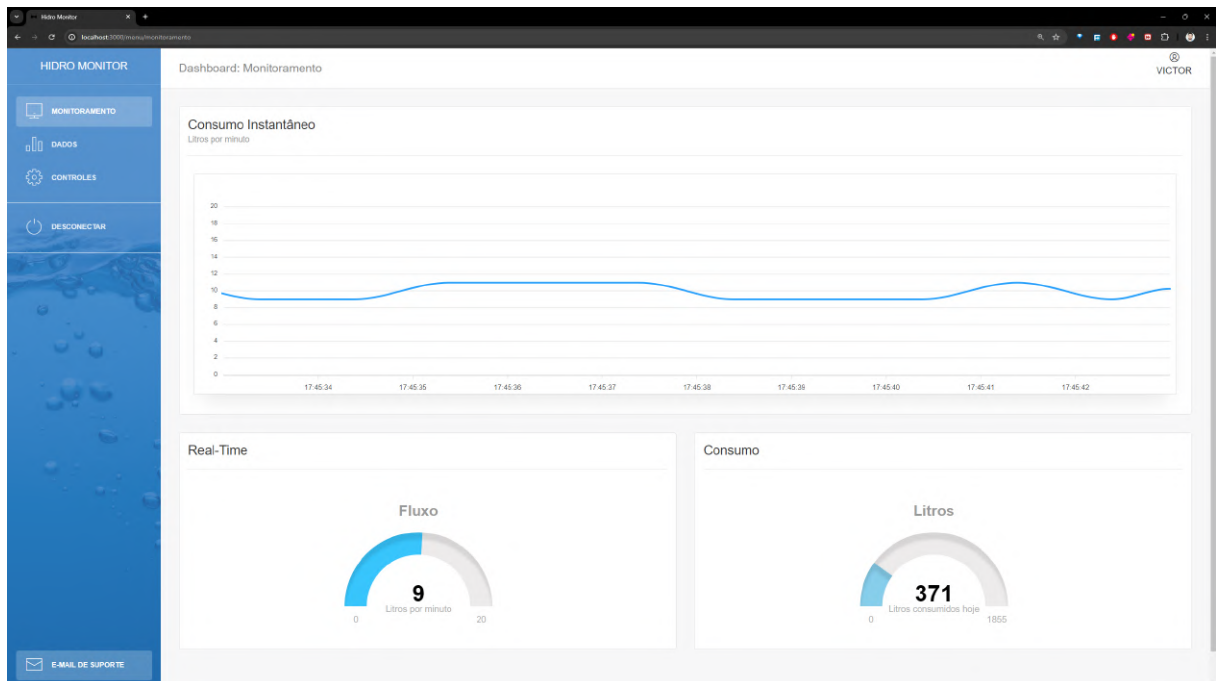
A fim de melhor validação no desempenho do equipamento, o protótipo do sistema foi instalado em um local de utilização comum, com o objetivo de testes similares à utilização real. Nas figuras 4.4 e 4.5 verifica-se o equipamento já instalado no mictório e pronto para a realização de testes.

Figura 4.4 – Protótipo instalado.



Fonte: Autor

Figura 4.5 – Protótipo instalado - monitoramento.

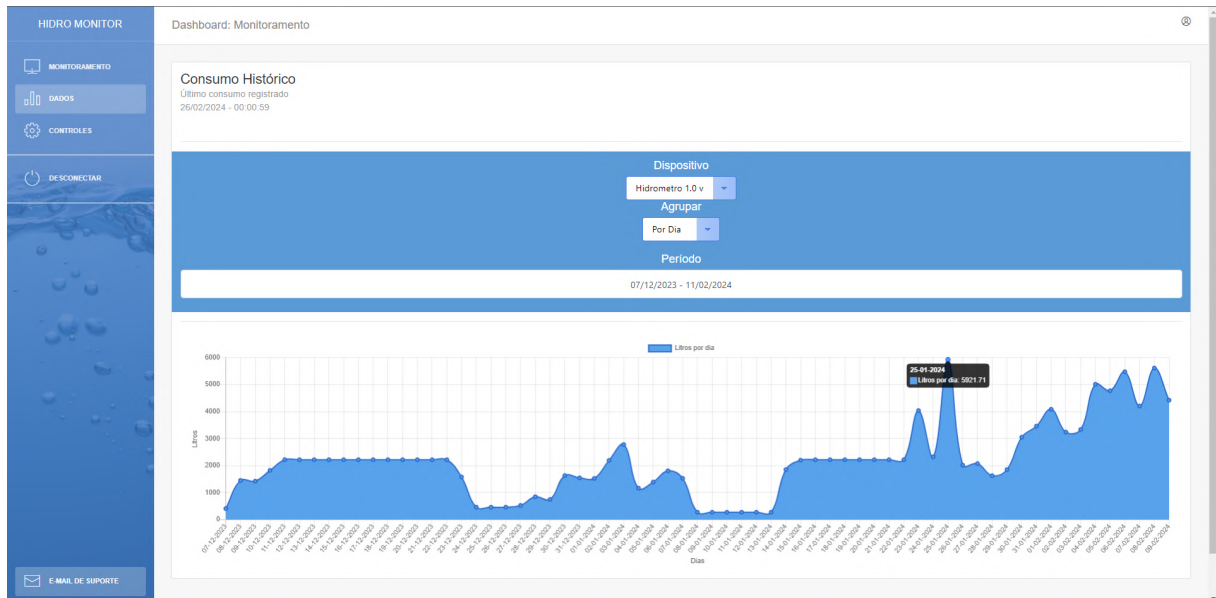


Fonte: Autor

Inicialmente, o equipamento foi instalado e desativado o acionamento automático de água; com esta configuração, o sistema realizou apenas o monitoramento do consumo

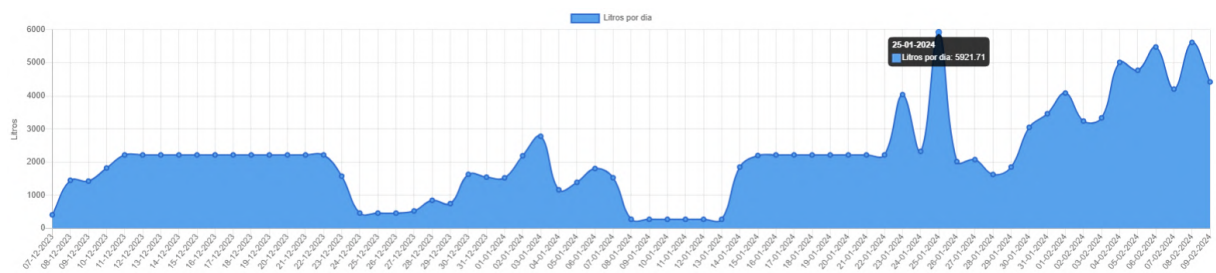
sem o uso do sistema de despejo de água. O equipamento ficou em configuração de monitoramento entre os dias 07/12/2023 e 09/02/2024. Durante esse período, o consumo foi armazenado na base de dados; nas figuras 4.6 e 4.7 mostram os gráficos de consumo diário dos dias monitorados.

Figura 4.6 – Consumo sem atuação do protótipo.



Fonte: Autor

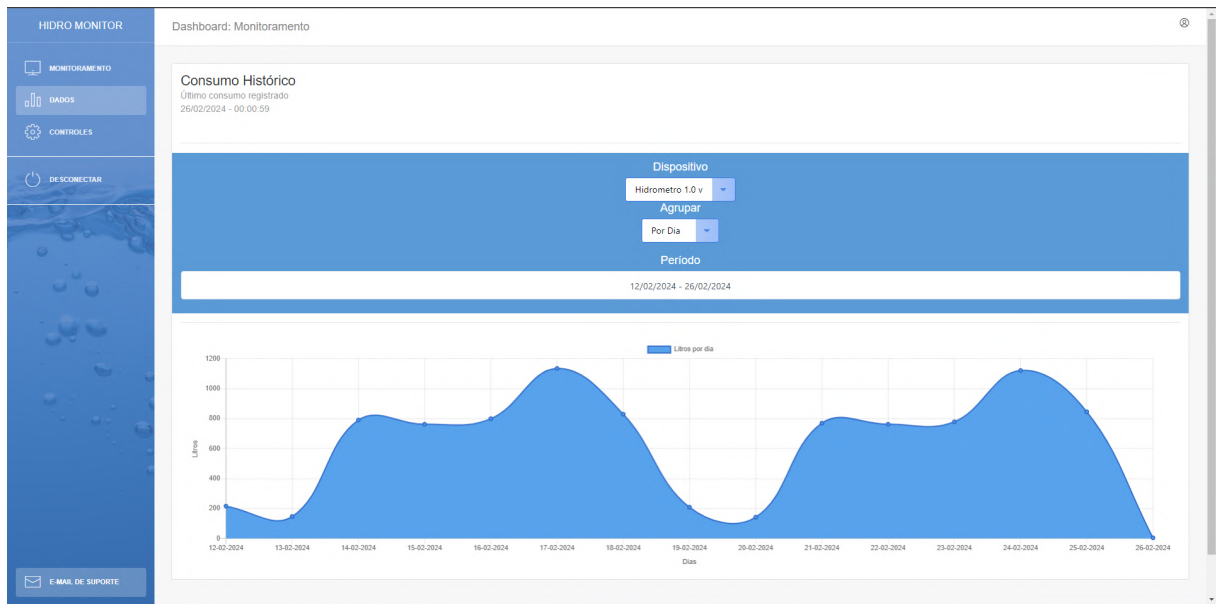
Figura 4.7 – Consumo sem atuação do protótipo - Ampliado.



Fonte: Autor

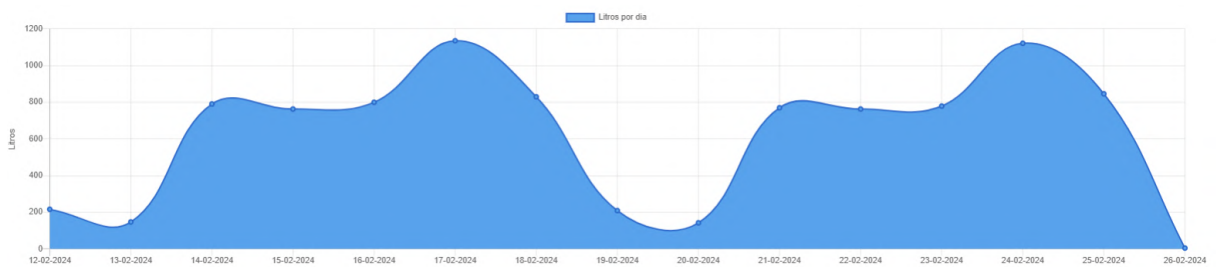
A segunda parte do teste no protótipo foi a ativação completa do sistema; deste modo, o equipamento pode trabalhar normalmente, detectando a utilização do mictório, acionando o despejo de água e simultaneamente registrando o consumo. Este teste foi realizado entre os dias 12/02/2024 e 25/02/2024. Nas Figuras 4.8 e 4.9 mostram o gráfico de consumo diário dos dias monitorados.

Figura 4.8 – Consumo com atuação do protótipo



Fonte: Autor

Figura 4.9 – Consumo com atuação do protótipo - Ampliado.



Fonte: Autor

Para realizar um comparativo entre os cenários monitorados, foram utilizados 13 dias sem operação e 13 dias com operação do protótipo. Ao observar a tabela 4.2 identifica-se uma redução no consumo de 52.218,91 litros para 9.311,49 litros, que é equivalente a uma redução de 82,16%, no consumo de água do local. Estes resultados mostram a eficiência do sistema em operação, demonstrando ser um sistema capaz de reduzir o desperdício de recursos e satisfazendo o principal objetivo do trabalho proposto.

Tabela 4.2 – Tabela de consumo mictório.

Sistema sem operação		Sistema em operação	
Dias	Consumo L	Dias	Consumo L
27/01/2024	2.076,11	12/02/2024	216,01
28/01/2024	1.627,22	13/02/2024	147,07
29/01/2024	1.850,85	14/02/2024	790,51
30/01/2024	3.049,96	15/02/2024	762,94
31/01/2024	3.459,78	16/02/2024	799,7
01/02/2024	4.084,80	17/02/2024	1.135,21
02/02/2024	3.239,31	18/02/2024	829,58
03/02/2024	3.335,85	19/02/2024	209,12
04/02/2024	5.011,87	20/02/2024	142,48
05/02/2024	4.771,39	21/02/2024	769,83
06/02/2024	5.474,45	22/02/2024	762,94
07/02/2024	4.203,81	23/02/2024	779,02
08/02/2024	5.613,11	24/02/2024	1121,42
09/02/2024	4.420,40	25/02/2024	845,66
Total	52.218,91	Total	9311,49

4.2.3 Banheiro masculino (UFPA Campus Tucuruí)

Com o intuito de monitorar e reduzir o consumo de água no campus, o projeto também foi implementado no campus universitário de Tucuruí, figuras 4.10, 4.11, 4.12; onde foi possível avaliar sua eficácia em um ambiente mais abrangente, monitorando e controlando o consumo em todo o banheiro. Além de comprovar a eficiência do sistema na gestão sustentável da água, a iniciativa resultou na entrega do projeto ao campus. Dessa forma, o sistema ficará acessível para os discentes, contribuindo para a educação ambiental e servindo como modelo de boas práticas para a comunidade acadêmica e para outros espaços públicos.

Figura 4.10 – Protótipo instalado.



Fonte: Autor

Figura 4.11 – Protótipo instalado - aberto.



Fonte: Autor

Figura 4.12 – Protótipo instalado - fechado.



Fonte: Autor

Para atender às necessidades específicas do ambiente, alguns parâmetros foram ajustados, como o tempo de fluxo de água permitido assim que uma presença é detectada, de modo a não interferir no uso das pias e no enchimento dos vasos sanitários. Além disso, o sensor de fluxo e a válvula solenoide foram instalados no encanamento externo aparente; também foi adicionado um desvio com registro, figura 4.13, como forma de redundância, para permitir o funcionamento do banheiro em caso de mau funcionamento do projeto.

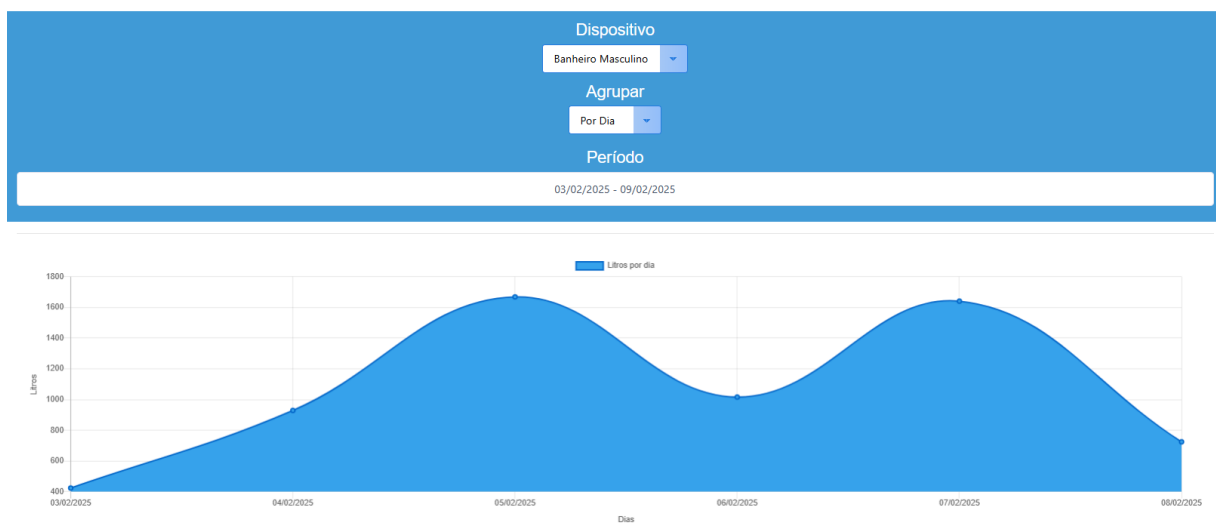
Figura 4.13 – Sensor de fluxo e solenoide instalados no encanamento do banheiro.



Fonte: Autor

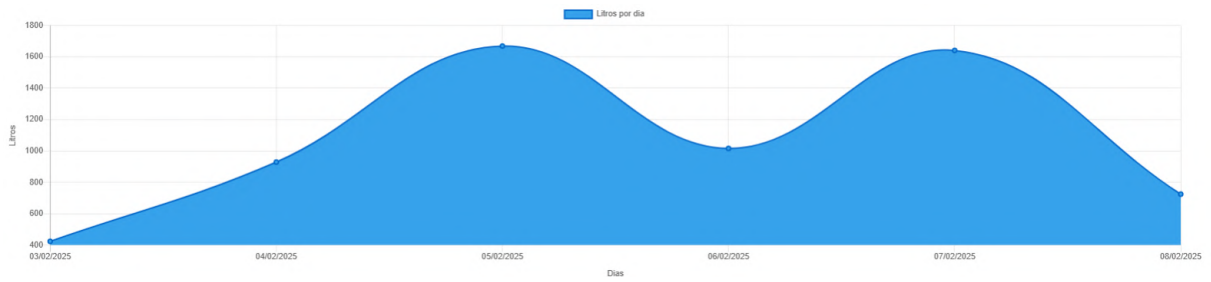
Figura 4.14 – Consumo com atuação do protótipo.

Consumo Histórico
Último consumo registrado
14/02/2025 - 20:08:48



Fonte: Autor

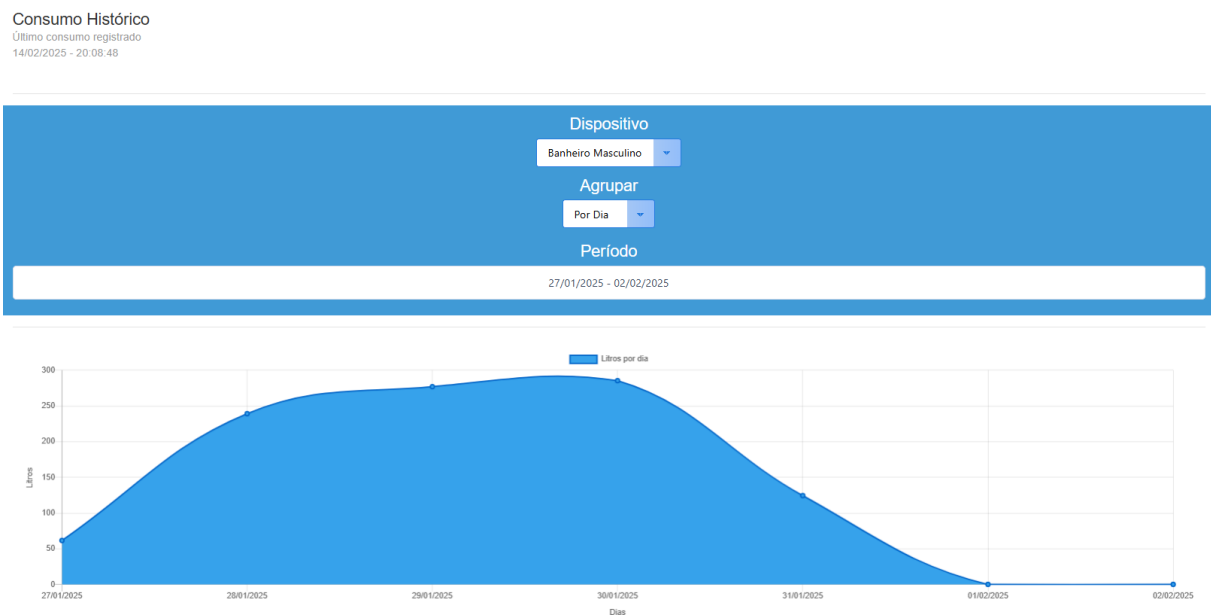
Figura 4.15 – Consumo com atuação do protótipo - Ampliado.



Fonte: Autor

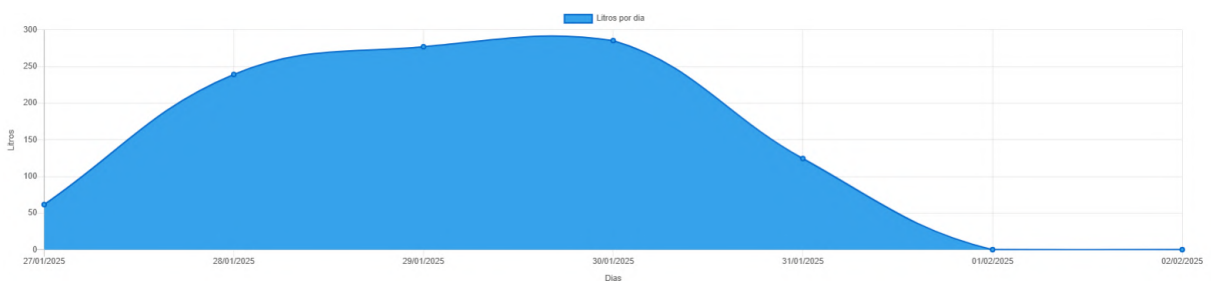
Em segundo momento, a detecção de presença foi desativada e o sistema passou apenas a monitorar o consumo sem interromper o fluxo hídrico. A data de início da amostragem foi 03/02/2025 e a de término 09/02/2025, observáveis nas figuras 4.16 e 4.17

Figura 4.16 – Consumo sem atuação do protótipo.



Fonte: Autor

Figura 4.17 – Consumo sem atuação do protótipo - Ampliado.



Fonte: Autor

Para podermos conferir os resultados obtidos, na primeira semana com o sistema atuante e habilitado para interromper o fluxo mediante a ausência de movimentos, comparamos com a segunda semana, onde o sistema passou somente a monitorar o consumo. Os dados foram coletados agrupados por dia conforme a tabela 4.3

Tabela 4.3 – Tabela de consumo banheiro (CAMTUC).

Sistema em operação		Sistema sem operação	
Dias	Consumo L	Dias	Consumo L
27/01/2025	61,68	03/02/2025	423,53
28/01/2025	239,19	04/02/2025	929,15
29/01/2025	277,00	05/02/2025	1.668,92
30/01/2025	285,04	06/02/2025	1.016,14
31/01/2025	124,49	07/02/2025	1.641,01
01/02/2025	0,02	08/02/2025	724,63
02/02/2025	0,02	09/02/2025	-
Total	987,44	Total	7.403,38

O consumo total na segunda semana foi de 7.403,38 litros, comparado a 987,44 litros na primeira semana, resultando em uma diferença de 6.415,94 litros. Em termos percentuais, essa diferença representa 649,62%, ou aproximadamente 7,49 vezes mais. Vale ressaltar que esse valor poderia ser ainda maior, já que não há dados para o dia 09/02/2025 devido à falta de água no campus.

4.3 Custo do Projeto

O projeto foi desenvolvido em duas partes principais: software e hardware. O software foi criado exclusivamente com ferramentas gratuitas, ocasionando a redução de custos de desenvolvimento. Já a construção física do projeto exigiu a aquisição de equipamentos, resultando em custos adicionais.

A Tabela 4.4 apresenta detalhadamente os valores dos seguintes componentes: ESP32, sensor de presença, sensor de fluxo, relé, válvula solenoide, regulador de tensão, fonte e placa de circuito impresso, totalizando R\$ 210,72. Esses custos podem variar conforme fatores como frete, fornecedor e marca dos equipamentos. Como os componentes foram adquiridos em 2020, são apresentados tanto os preços da época quanto os valores atualizados.

O total foi considerado adequado, levando em conta a ampla variedade de possíveis aplicações e os resultados alcançados por este projeto. Além disso, a expectativa de durabilidade é elevada, o que reforça seus benefícios a longo prazo.

Tabela 4.4 – Tabela de custos do protótipo do hardware.

Equipamentos	Valor unitário	Qtd	Total (02/2020)	Total (10/2024)
Esp32	R\$ 39,00	1	R\$ 28,00	R\$ 39,00
Sensor de Presença	R\$ 8,45	1	R\$ 6,78	R\$ 8,45
Sensor de Fluxo	R\$ 38,37	1	R\$ 29,29	R\$ 35,91
Relé	R\$ 11,00	1	R\$ 8,38	R\$ 11,00
Válvula Solenóide	R\$ 59,00	1	R\$ 39,00	R\$ 44,90
Regulador de Tensão	R\$ 28,90	1	R\$ 19,25	R\$ 28,90
Fonte 12V 2A	R\$ 15,90	1	R\$ 15,90	R\$ 21,90
Placa de Circuito Impresso	R\$ 26,00	1	R\$ 26,00	R\$ 33,17
Valor total			R\$ 172,60	R\$ 223,23

5 CONCLUSÃO

O protótipo do sistema de controle e monitoramento demonstrou-se uma solução eficiente e robusta para a gestão do consumo de água, especialmente em ambientes como banheiros públicos e privados. A criação de um equipamento autônomo que não só reduz o consumo excessivo de água, mas também permite o monitoramento em tempo real e o armazenamento de dados, mostrou-se uma ferramenta poderosa para análises e possíveis adaptações para outras necessidades e aplicações, abrindo o leque para as mais variadas aplicações.

A complexidade do sistema, que envolve hardware para acionamento automático e monitoramento hídrico, um banco de dados robusto para o armazenamento das informações, e um servidor web que facilita a interface com o usuário, comprova a viabilidade e a eficiência da solução proposta. Através dos testes realizados, verificou-se a funcionalidade e a confiabilidade do sistema, validando a proposta inicial do projeto.

A integração entre o hardware e o software foi essencial para garantir a operação do sistema, onde o uso do PostgreSQL para gerenciamento de dados e a implementação de uma interface web em Node.js podem proporcionar uma experiência intuitiva e funcional para o usuário. Este projeto exemplifica como a tecnologia pode ser aplicada de maneira prática e eficaz para melhorar a gestão e a operação de sistemas automatizados em diferentes contextos, sejam eles residenciais, comerciais ou até mesmo industriais.

Em resumo, o sistema de controle e monitoramento de vazão apresenta uma solução completa e escalável, capaz de atender às demandas diárias de sustentabilidade e eficiência no uso de recursos hídricos. O projeto não só atinge seus objetivos iniciais como também abre caminho para futuras implementações e melhorias.

A viabilidade econômica do projeto é reforçada pela análise de custo-benefício, onde os investimentos iniciais em hardware e software são compensados pelas economias geradas na conta de água. Ademais, a capacidade de armazenamento e análise de dados históricos permite uma gestão proativa, identificando padrões de consumo e possibilitando a implementação de estratégias de uso eficiente da água.

5.1 Trabalhos Futuros

Para maximizar os benefícios econômicos do sistema desenvolvido, sugere-se:

- Integração com sistemas de Internet das Coisas (IoT) para ampliar a conectividade e o controle remoto, potencializando a economia.
- Implementação de algoritmos de aprendizado de máquina para análise preditiva do consumo de água, permitindo ajustes automáticos e mais eficientes.

- Desenvolvimento de uma aplicação mobile para facilitar o acesso e o monitoramento dos dados pelos usuários, promovendo uma gestão mais ativa e consciente.
- Expansão do sistema para incluir outros tipos de sensores, possibilitando ampliar sua aplicação no monitoramento de consumo em diferentes ambientes, como regular o consumo total de uma casa, prédios comerciais ou até mesmo em escala industrial, onde poderia atuar também como agente promotor de segurança, monitorando e interrompendo o fluxo de gases e/ou líquidos em uma cadeia produtiva, diante de uma anomalia no fluxo, aumentando assim sua abrangência e impacto econômico.

REFERÊNCIAS

- ABBATE, J. E. **From ARPANET to Internet: A history of ARPA-sponsored computer networks, 1966-1988**. [S.l.]: University of Pennsylvania, 1994. Citado na página 27.
- ALBUQUERQUE, Y. **ESP32 pinout – Guia Básico de GPIOs**. 2020. <<https://blog.smartkits.com.br/esp32-pinout-guia-basico-de-gpios/>>. Citado 2 vezes nas páginas 18 e 19.
- ALMEIDA, V.; COSTA, F. Integração de sistemas de monitoramento de água com plataformas web: Desafios e soluções. **International Journal of Environmental Technology**, v. 14, n. 4, p. 78–90, 2018. Citado na página 14.
- ARDUINO. **Arduino**. 2022. <<https://www.arduino.cc/en/about/>>. Citado na página 16.
- BRASIL, W. **Falta de água potável preocupa 81% dos brasileiros, aponta estudo**. 2023. Acesso em: 04 fev. 2025. Disponível em: <<https://www.wwf.org.br/?85100/Falta-de-agua-potavel-preocupa-81-dos-brasileiros-aponta-estudo>>. Citado na página 13.
- CITISYSTEMS. **Válvula Solenoide**. 2019. Acesso em: 15 ago. 2020. Disponível em: <<https://citisystems.com.br>>. Citado na página 23.
- CLIMAINFO. **Demanda por água pode aumentar quase 80% até 2040 no Brasil**. 2020. Acesso em: 04 fev. 2025. Disponível em: <<https://climainfo.org.br/2020/08/27/demanda-por-agua-pode-aumentar-quase-80-ate-2040-no-brasil/>>. Citado na página 13.
- COODESH. **O que é NoSQL?** 2020. <<https://coodesh.com/blog/dicionario/o-que-e-nosql/>> [Acessado em: 18/11/2020]. Citado na página 29.
- CROMATEK. **Estrutura de um Relé**. 2021. Acesso em: 29 mar. 2022. Disponível em: <<https://cromatek.com.br/wp-content/uploads/2021/03/estrutura-do-rele.jpg>>. Citado 2 vezes nas páginas 16 e 20.
- DUMBO. **Osciloscópio mostrando pulsos de um sensor de fluxo com sensor Hall**. 2018. Electrical Engineering Stack Exchange. Acesso em: 20 set. 2019. Disponível em: <<https://electronics.stackexchange.com/questions/393983/dealing-with-coupling-of-pwm-noise-into-hall-sensors-as-spikes-in-blcd-driver-ci>>. Citado na página 24.
- ECMA. **JavaScript**. 2024. <<https://www.ecma-international.org/publications/standards/Ecma-262.htm>>. Citado na página 16.
- ELETRONICS, A. Relé-o que é e como funciona. **Acesso em**, v. 17, 2019. Citado na página 20.
- ELETRONICS, A. Funcionamento e aplicações do sensor pir. **Revista de Engenharia Eletrônica**, v. 15, n. 2, p. 34–47, 2020. Citado na página 21.
- ESPRESSIF. **ESP32 Series Datasheet**. 2020. Accessed: 2021-03-29. Disponível em: <https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf>. Citado na página 18.

ESPRESSIF. **ESP32**. 2022. <<https://www.espressif.com/en/products/socs/esp32>>. Citado na página 16.

EXPRESSJS.COM. **Express - Fast, unopinionated, minimalist web framework for Node.js**. 2024. <<https://expressjs.com/>>. Citado na página 27.

G1 Meio Ambiente. **Relatório da ONU alerta para escassez de água global**. 2023. Acesso em: 01 ago. 2023. Disponível em: <<https://g1.globo.com/meio-ambiente/noticia/2023/03/21/relatorio-da-onu-aponta-risco-de-crise-global-por-escassez-de-agua.ghtml>>. Citado na página 13.

GIOVANNI.DE.CASTRO. **Introdução ao Regulador de Tensão**. 2024. Acessado em: 09 abr. 2024. Disponível em: <<https://www.robocore.net/tutoriais/introducao-regulador-de-tensao>>. Citado na página 25.

MICROSOFT. **Visual Studio Code**. 2023. <<https://code.visualstudio.com/docs>>. Citado na página 16.

MONGODB. **MongoDB – Banco de Dados NoSQL**. 2020. <<https://www.mongodb.com>> [Acessado em: 18/11/2020]. Citado na página 29.

NODEJS.ORG. **Node.js - Execute JavaScript em qualquer lugar**. 2024. <<https://nodejs.org/pt/about>>. Citado 2 vezes nas páginas 16 e 27.

OLIVEIRA, R. Sensores de movimento e suas aplicações em automação residencial. **Journal de Automação Inteligente**, v. 10, n. 1, p. 22–30, 2019. Citado na página 22.

POSTGRESQL. **PostgreSQL**. 2023. <<https://www.postgresql.org/>>. Citado na página 16.

PRESSMAN, R. S. **Engenharia de Software**. 6. ed. São Paulo: McGrawHill, 2006. Citado na página 26.

RAYZEEK. **Sensor de Movimento PIR**. 2025. <<https://www.rayzeek.com/pt/blog/sensor-de-movimento-pir>>. Citado na página 16.

ROBOCORE. **Sensor de Presença PIR HC-SR501**. 2024. Acesso em: 04 out. 2024. Disponível em: <<https://www.robocore.net/sensor-ambiente/sensor-de-presenca-pir-hc-sr501>>. Citado na página 22.

SILVA, J.; OLIVEIRA, R.; MARTINEZ, A. Automação no controle do consumo de água em ambientes industriais e residenciais. **Revista de Engenharia e Tecnologia**, v. 12, n. 3, p. 45–58, 2020. Citado na página 13.

SILVEIRA, C. B. **Como Funciona a Válvula Solenoide e Quais os Tipos?** 2017. <<https://www.citisystems.com.br/valvula-solenoide/>> [Acessado em: 18/01/2024]. Citado na página 22.

SILÍCIO, V. de. **Sensor de Presença HC-SR501**. 2020. Acesso em: 02 jan. 2021. Disponível em: <<https://portal.vidadesilicio.com.br/sensor-de-presenca-hc-sr501/>>. Citado na página 21.

SOARES, A. E. P.; PRADO, A. R. M. do; SILVA, S. R. da. O monitoramento como ferramenta da redução do consumo de água potável na faculdade de ciências da administração de pernambuco – fcap/upe. **Tecno-Lógica**, v. 23, n. 1, p. 42–48, 2019. Disponível em: <<https://online.unisc.br/seer/index.php/tecnologica/article/view/11693>>. Citado 2 vezes nas páginas 14 e 17.

UNIDAS, O. das N. **Relatório da ONU aponta risco de crise global por escassez de água**. 2023. Acesso em: 04 fev. 2025. Disponível em: <<https://g1.globo.com/meio-ambiente/noticia/2023/03/21/relatorio-da-onu-aponta-risco-de-crise-global-por-escassez-de-agua.ghtml>>. Citado na página 13.

WEBSTER, J. G.; EREN, H. **Measurement, instrumentation, and sensors handbook: spatial, mechanical, thermal, and radiation measurement**. [S.l.]: CRC press, 2017. Citado na página 24.

WIKIPEDIA. **Hall Effect Sensor**. 2024. <https://en.wikipedia.org/wiki/Hall_effect_sensor>. Citado na página 16.

WIKIPEDIA. **Solenoid Valve**. 2024. <https://en.wikipedia.org/wiki/Solenoid_valve>. Citado na página 16.

WOLF, W. **Computers as Components: Principles of Embedded Computing System Design**. [S.l.]: Morgan Kaufmann, 2012. Citado na página 17.

APÊNDICE A – Repositórios no GitHub

Os repositórios utilizados neste projeto estão disponíveis nos seguintes links:

- Hidrometro ESP32
- Hidrometro NodeJs