



Universidade Federal de Pernambuco

Departamento de Eletrônica e Sistemas

Curso de Engenharia Eletrônica

**Sistema para gerenciamento e controle do consumo hidráulico
de residências baseado em ESP32**

Trabalho de Conclusão de Curso de Graduação

por

Dayvison Pedro Pilar da Silva

Orientador: João Marcelo Teixeira

Recife, Julho de 2023

Dayvison Pedro Pilar da Silva

**Sistema para gerenciamento e controle do consumo hidráulico de residências
baseado em ESP32**

Monografia apresentada ao Curso de Engenharia Eletrônica, como requisito parcial para a obtenção do Título de Bacharel em Engenharia Eletrônica, do Centro de Tecnologia e Geociências da Universidade Federal de Pernambuco.

Orientador: João Marcelo Teixeira

Recife

2023

DAYVISON PEDRO PILAR DA SILVA

**SISTEMA PARA GERENCIAMENTO E CONTROLE DO CONSUMO HIDRÁULICO
DE RESIDÊNCIAS BASEADO EM ESP32**

Trabalho de Conclusão de Curso
apresentado ao Curso de Graduação em
Engenharia Eletrônica da Universidade
Federal de Pernambuco, Centro de
Tecnologia e Geociências, como requisito
parcial para a obtenção do grau de
Bacharel em Engenharia Eletrônica.

Aprovado em: 02/05/2023.

BANCA EXAMINADORA

Prof. Dr. João Marcelo Xavier Natário Teixeira (Orientador)
Universidade Federal de Pernambuco

Prof. Dr. Saulo Oliveira Dornellas Luiz (Examinador Interno)
Universidade Federal de Pernambuco

Ficha de identificação da obra elaborada pelo autor,
através do programa de geração automática do SIB/UFPE

Silva, Dayvison Pedro Pilar da .

Sistema para gerenciamento e controle do consumo hidráulico de residências baseado em ESP32 / Dayvison Pedro Pilar da Silva. - Recife, 2023.

53 f. : il., tab.

Orientador(a): João Marcelo Xavier Natário Texeira

Cooorientador(a): Saulo Oliveira Dornellas Luiz

Trabalho de Conclusão de Curso (Graduação) - Universidade Federal de Pernambuco, Centro de Tecnologia e Geociências, Engenharia Eletrônica - Bacharelado, 2023.

1. IOT. 2. Eletrônica. I. Texeira, João Marcelo Xavier Natário. (Orientação).
II. Luiz, Saulo Oliveira Dornellas. (Coorientação). III. Título.

600 CDD (22.ed.)

Agradecimentos

A toda a minha família que, com muito carinho e apoio, não mediram esforços para que eu chegasse até esta etapa de minha vida.

Aos amigos que fiz durante o curso, particularmente Jorge, Renato, Lucas, Gabriel, Thiago. Sem vocês, não seria possível ver a reta final deste desafio.

A todos os professores do Departamento de Eletrônica e Sistemas, mas principalmente ao professor João Marcelo, que se colocou à disposição de forma irrestrita para que eu pudesse realizar este projeto, mesmo nas situações mais adversas.

*O dom de poder mental vem de Deus,
o ser Divino e se concentrarmos nossas mentes na verdade,
ficamos em sintonia com este grande poder.*

Nikola Tesla

RESUMO

A Internet das Coisas (IoT) refere-se a uma rede de dispositivos físicos, veículos, aparelhos e outros objetos físicos que são incorporados com sensores, software e conectividade de rede que lhes permite coletar e compartilhar dados. Esses dispositivos – também conhecidos como “objetos inteligentes” – podem variar de dispositivos simples de “casa inteligente”, como termostatos inteligentes, a vestíveis como smartwatches e roupas habilitadas para RFID, até máquinas industriais complexas e sistemas de transporte. Os tecnólogos estão até imaginando “cidades inteligentes” inteiras baseadas em tecnologias IoT.

A IoT permite que esses dispositivos inteligentes se comuniquem entre si e com outros dispositivos habilitados para internet, como smartphones e gateways, criando uma vasta rede de dispositivos interconectados que podem trocar dados e executar uma variedade de tarefas de forma autônoma. Isso pode incluir tudo, desde o monitoramento das condições ambientais em fazendas até o gerenciamento de padrões de tráfego com carros inteligentes e outros dispositivos automotivos inteligentes, controle de máquinas e processos em fábricas e rastreamento de estoque e remessas em armazéns.

As aplicações potenciais da IoT são vastas e variadas, e seu impacto já está sendo sentido em uma ampla gama de setores, incluindo manufatura, transporte, saúde e agricultura. À medida que o número de dispositivos conectados à Internet continua a crescer, é provável que a IoT desempenhe um papel cada vez mais importante na formação de nosso mundo e na transformação da maneira como vivemos, trabalhamos e interagimos uns com os outros.

Em um contexto empresarial, os dispositivos IoT são usados para monitorar uma ampla gama de parâmetros, como temperatura, umidade, qualidade do ar, consumo de energia e desempenho da máquina. Esses dados podem ser analisados em tempo real para identificar padrões, tendências e anomalias que podem ajudar as empresas a otimizar suas operações e melhorar seus resultados.

Palavras-chave: IOT e Eletrônica.

ABSTRACT

The Internet of Things (IoT) refers to a network of physical devices, vehicles, appliances and other physical objects that are embedded with sensors, software and network connectivity that allows them to collect and share data. These devices — also known as “smart objects” — can range from simple “smart home” devices like smart thermostats, to wearables like smartwatches and RFID-enabled clothing, to complex industrial machinery and transportation systems. Technologists are even envisioning entire “smart cities” predicated on IoT technologies.

IoT enables these smart devices to communicate with each other and with other internet-enabled devices, like smartphones and gateways, creating a vast network of interconnected devices that can exchange data and perform a variety of tasks autonomously. This can include everything from monitoring environmental conditions in farms, to managing traffic patterns with smart cars and other smart automotive devices, to controlling machines and processes in factories, to tracking inventory and shipments in warehouses.

The potential applications of IoT are vast and varied, and its impact is already being felt across a wide range of industries, including manufacturing, transportation, healthcare and agriculture. As the number of internet-connected devices continues to grow, IoT is likely to play an increasingly important role in shaping our world and transforming the way we live, work and interact with each other.

In an enterprise context, IoT devices are used to monitor a wide range of parameters such as temperature, humidity, air quality, energy consumption and machine performance. This data can be analyzed in real time to identify patterns, trends and anomalies that can help businesses optimize their operations and improve their bottom line.

Key-words: IOT and Eletronic.

LISTA DE FIGURAS

Figura 1	Tabela de distribuição de água no Brasil por região.	12
Figura 2	Dias de abastecimento de água(Arthur Lundgren II).	13
Figura 3	Dias de abastecimento de água(Rio Doce).	14
Figura 4	PCB do sistema proposto <i>EasyEDA(Standard)</i>	17
Figura 5	PCB Layout.	18
Figura 6	Esp32 Wroom Devkit v1.	19
Figura 7	Fotografia da top Layer da PCB.	20
Figura 8	Fotografia da protoboard do projeto.	21
Figura 9	Fotografia do Sensor de Fluxo de Água 1/2 YF-S201b.	22
Figura 10	Sensor de Fluxo de Água - Interno.	23
Figura 11	Fotografia da bóia com contrapeso.	23
Figura 12	Relé interno à bóia [1].	24
Figura 13	Esquemas de ligação da bóia [1]	24
Figura 14	Esquemas de ligação da bóia [2]	25
Figura 15	Esquema elétrico.	26
Figura 16	Esquema elétrico - ESP32.	26
Figura 17	Esquema elétrico - Tensões.	27
Figura 18	Esquema elétrico - Sensor de fluxo.	27
Figura 19	Esquema elétrico - Bóia.	28
Figura 20	Esquema elétrico - Relé.	28
Figura 21	Comunicação entre o dispositivo móvel, servidor do <i>blynk</i> e o <i>esp32</i> [3]. ..	30
Figura 22	Criação do projeto na plataforma <i>blynk</i>	31
Figura 23	Caixa de Widget do <i>blynk</i>	32
Figura 24	<i>LED</i> virtual - Boia.	32
Figura 25	<i>LED</i> virtual - Sensor de fluxo.	33
Figura 26	Display da vazão de água.	33
Figura 27	<i>LED</i> virtual do relé.	34
Figura 28	Painel de eventos com as variáveis.	35
Figura 29	Painel de eventos do <i>blynk</i>	36

Figura 30	Led física da boia.....	37
Figura 31	Gerenciador dos Widgets.	38
Figura 32	Led física do sensor de fluxo.....	39
Figura 33	Led virtual do fluxo.	39
Figura 34	Led física do relé.	40
Figura 35	Led virtual do relé.....	40
Figura 36	PCB fixada no local dos testes.	41
Figura 37	Instalação do fio da bóia na PCB.	42
Figura 38	Fio da bóia dentro da caixa d'água.	42
Figura 39	Bóia dentro da caixa d'água.	43
Figura 40	<i>LED</i> indicadora de nível baixo na caixa.	43
Figura 41	Hora que o nível baixo de água ficou baixo no painel de eventos.	44
Figura 42	Sensor de fluxo instalado na encanação.....	44
Figura 43	Sistema funcionando na PCB e no aplicativo.	45
Figura 44	Data e hora em que chegou água na rua.....	45
Figura 45	Ligação elétrica do AC da bomba com o relé.....	46
Figura 46	Enchimento da caixa d'água.....	47
Figura 47	Boia desligando o sistema devido ao nível alto de água na caixa.	47

LISTA DE TABELAS

Tabela 1	Preços estimados para componentes descritos na solução.	29
----------	--	----

LISTA DE SIGLAS

PCB	Printed Circuit Board
UFPE	Universidade Federal de Pernambuco
IOT	Internet Of Things
ESP32	Sistema Embarcado da Espressif
BLYNK	Aplicativo de Desenvolvimento
IDE	Ambiente de Desenvolvimento Integrado
PROTOBOARD	Placa de Teste de Circuitos
LAYOUT	Design das Placas
PADS	Ilhas de Soldagem Eletrônica
LAYERS	Camadas de Uma Placa de Circuito Impresso
WIDGETS	Atalhos De Um Aplicativo
EasyEDA(Standard)	Software de Desenvolvimento de Placas
JLCPCB	Empresa de Prototipagem de Placas Eletrônicas
LABELS	Sinalização de um Tipo de Sinal
OUTPUT	Sinais de Saída
INPUT	Sinais de Entrada
VCC	Alimentação circuito integrado
GND	Terra
E/S	Entrada/Saída
SILK SCREEN	Escritas na Placa de Circuito
V	Tensão

SUMÁRIO

1	INTRODUÇÃO	12
1.1	Objetivos	13
1.1.1	Objetivo Geral	13
1.1.2	Objetivos Específicos	14
1.2	Organização do texto	14
2	TRABALHOS RELACIONADOS	16
3	SOLUÇÃO PROPOSTA	17
3.1	Hardware	18
3.1.1	Esp32 Wroom Devkit v1	18
3.1.2	PCB e Protoboard	20
3.1.3	Sensor de fluxo	22
3.1.4	Boia de nível	23
3.1.4.1	Funcionamento da bóia	24
3.1.5	Relé	25
3.1.6	Esquema elétrico	26
3.1.7	Estimativa de Orçamento	29
3.2	Software	29
3.2.1	Linguagem Arduino Sketch	29
3.2.2	Ferramenta <i>blynk</i>	30
3.2.3	Configuração da plataforma <i>blynk</i>	31
3.2.4	Interface e <i>widgets</i> do <i>blynk</i>	31
3.2.5	Painel de eventos	35
3.3	Funcionamento do projeto	37
4	TESTES	41
4.1	Local dos testes	41
4.1.1	Instalação da boia	42
4.1.2	Instalação do sensor de fluxo	44
4.1.3	Instalação do relé com a bomba	46

5	DIFICULDADES ENCONTRADAS	48
6	CONCLUSÃO E TRABALHOS FUTUROS	49

1 INTRODUÇÃO

Cerca de 70% da superfície do planeta Terra é composta por água (PENA; RODOLFO, 2000) [4]. No entanto, apenas 2,6% dessa água é doce, distribuída em sua maioria em geleiras nas calotas polares e em águas subterrâneas, restando uma parcela de 0,3% do total de água do planeta acessível ao ser humano (SOUSA; SARDINHA, 2019) [5]. Além de ser um recurso limitado, a água potável não é bem distribuída ao redor do mundo. Ainda de acordo com Sousa e Sardinha, 60% do recurso disponível está concentrado em apenas 9 países: Canadá, EUA, Colômbia, China, Rússia, Indonésia, Índia, Congo e Brasil.

O Brasil, embora contenha 12% de toda a água doce do planeta, também tem essa riqueza mal distribuída em seu território. A maior parte é disposta na região norte, sendo as regiões sudeste, sul e nordeste possuidoras de poucos recursos hídricos. Essa distribuição gera uma problemática, pois o norte possui baixa densidade demográfica e a maior parte da população brasileira se concentra na região sudeste, como mostra a Figura 1 abaixo.

Disponibilidade hídrica per capita (m ³ /hab/ano)	Estados	Situação
> 20.000	AC, AM, AP, GO, MS, MT, PA, RO, RR, RS e TO	Riquíssimo
> 10.000	MA, MG, SC e PR	Muito Rico
> 5.000	ES e PI	Rico
> 2.500	BA e SP	Situação adequada
< 2.500	CE, RJ, RN, DF, AL e SE	Pobres
< 1.500	PB e PE	Situação crítica

Figura 1: Tabela de distribuição de água no Brasil por região.
[6]

Na última década, várias localidades do nosso país tiveram um aumento nos problemas com a falta d'água, onde diversas cidades encararam racionamentos anualmente. Alguns fatores como poluição em rios, desmatamento, alteração do ciclo das chuvas e má gestão na utilização dos recurso hídricos têm agravado a situação. O saneamento básico do Brasil ainda é bastante ineficiente, por exemplo. Apenas 46% do esgoto do país recebe tratamento, o restante é despejado na natureza, em grande maioria nos rios, aumentando a probabilidade de proliferação de doenças, contaminação da água e consequentemente diminuindo a sua disponibilidade potável (BRKAMBIENTAL, 2020) [7]. Outro dado im-

portante que esta fonte traz é sobre o tratamento de esgoto nas 90 maiores cidades do país. Apenas 22 dessas cidades tem mais de 80% da sua população recebendo coleta de esgoto, contribuindo para que 80% dessas grandes cidades tenham uma perda de cerca de 30% de água potável em sua distribuição.

Em particular, a região metropolitana do Recife (Nordeste) sofre com o sistema de distribuição de água, gerenciada pela empresa de abastecimento(Compesa) [8], podemos vizualizar tal escassez no município de Arthur Lundgren II(Paulista), no qual o calendário da Compesa na Figura 2 mostra um dia parcial de água semanal, ou seja, deve ser feito o armazenamento dessa água nas caixas d'água neste dia , caso contrário não terão água para utilizar durante a semana.Semelhantemente a este cenário temos a região de Rio Doce onde o abastecimento é totalmente precário, e cujo calendário de abastecimento na Figura 3.

Associado a este problema temos também as grandes perdas de água, pela necessidade de um acionamento manual da bomba d'água por longo intervalos de tempo, no qual a caixa d'água está cheia e transborda litros de água pela necessidade do usuário de desligar e ligá-la manualmente.

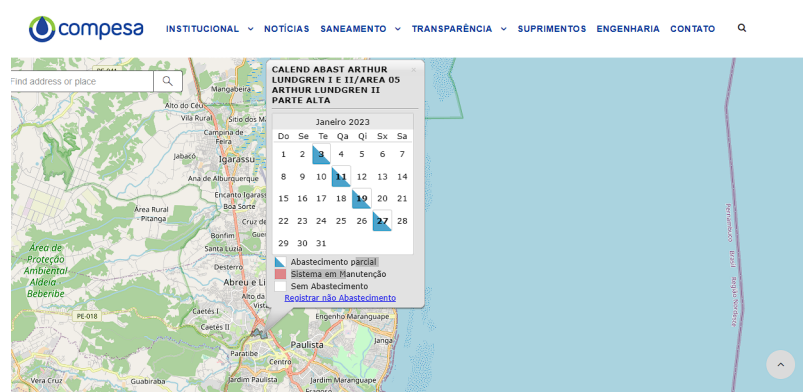


Figura 2: Dias de abastecimento de água(Arthur Lundgren II).

1.1 Objetivos

1.1.1 Objetivo Geral

Automatizar o enchimento de caixa d'água em residências nas quais o abastecimento é precário, realizando seu monitoramento, sempre visando a menor perda de água no seu enchimento e todo o seu monitoramento feito por software.

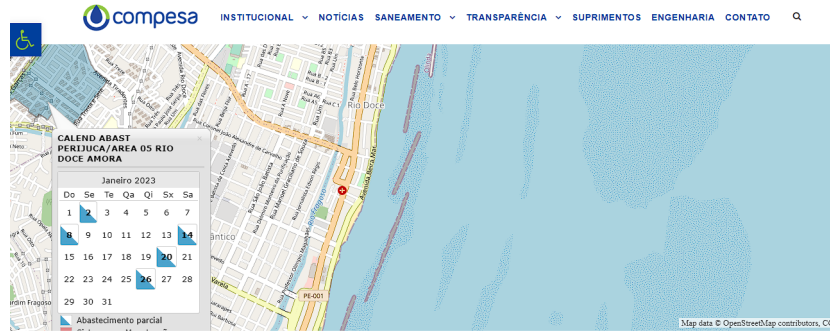


Figura 3: Dias de abastecimento de água(Rio Doce).

1.1.2 Objetivos Específicos

Para atingir o objetivo geral, foram definidos os seguintes objetivos específicos:

- Desenvolver uma PCB que faça todo o monitoramento de fluxo de água residencial
- Desenvolver um aplicativo móvel acessível por rede sem fio.
- Utilizar o *firmware* DEV KIT, o *blynk* e suas bibliotecas
- Utilizar linguagem Arduino para o desenvolvimento do projeto

1.2 Organização do texto

Este trabalho está dividido em seis capítulos, os quais abordam a contextualização e motivação à realização desse trabalho, assim como os principais trabalhos relacionados, o desenvolvimento do trabalho em si, assim como os testes realizados, principais dificuldades encontradas e as conclusões obtidas com o mesmo. A separação utilizada para os capítulos é explicada abaixo, assim como os tópicos que cada um contém:

- Capítulo 1: Introdução, engloba a contextualização do trabalho e seus objetivos.
- Capítulo 2: Trabalhos Relacionados, lista alguns trabalhos encontrados que se relacionam diretamente com a presente proposta.
- Capítulo 3: Solução proposta, traz a definição do projeto desenvolvido nesse trabalho e detalha as tecnologias nele utilizadas, juntamente com o detalhamento do desenvolvimento realizado.
- Capítulo 4: Testes, detalha a execução dos testes de validação da proposta.

- Capítulo 5: Dificuldades Encontradas, lista as dificuldades encontradas nesse trabalho e como elas foram superadas.
- Capítulo 6: Conclusão e Trabalhos Futuros, apresenta as conclusões acerca do trabalho desenvolvido e lista possibilidades futuras para continuidade do mesmo.

2 TRABALHOS RELACIONADOS

Com o interesse crescente nos últimos tempos, e devido à complexidade e à possibilidade de aprendizado, a implementação de sistemas automatizados inteligentes que envolve IOT [9] tem crescido bastante pois explora bastante os conhecimentos de *hardware* e *software*, principalmente microcontroladores. Em seguida, serão listados alguns projetos com uma premissa de criar uma PCB para automatização de acionamento de uma bomba d'água, com diferentes abordagens.

No artigo de (DORNELAS) [10], foi apresentado um protótipo bastante parecido com o proposto neste trabalho. Nele, foi desenvolvido um sistema de monitoramento de consumo de água utilizando o próprio Arduino com um sensor de fluxo, mas a plataforma de comunicação para acompanhamento em tempo real do consumo foi desenvolvida utilizando-se um *Raspberry Pi* e um rádio nRF24L01+, que fazem a coleta dos dados e enviam para o software desenvolvido em KNoT, onde o usuário possui acesso em tempo real às informações.

Em outra aplicação semelhante, (OLIVEIRA; SANTOS; RODRIGUES, 2014) [11] utilizou um Arduino Mega 2560, um sensor de pressão MPX5010dp e um módulo *bluetooth* RS 232 HC-05 para monitorar o nível de água de uma caixa. O sensor de pressão é responsável pela leitura dos dados e envio ao Arduino. No programa do microcontrolador, um *software* foi desenvolvido para fazer a leitura da pressão recebida e por meio de cálculos transformá-la em volume de água. Esse volume é enviado para um aplicativo desenvolvido em plataforma Android, através da conexão do dispositivo com o módulo *bluetooth*, que está interligado ao controlador.

Os projetos acima citados, inspiram e auxiliam a execução do modelo apresentado neste trabalho. O dispositivo de medição, sensor de fluxo, é de um modelo parecido com o utilizado no projeto de (DORNELAS; OLIVEIRA, 2017) [10]. O desenvolvimento do *software* e a comunicação entre microcontrolador e ele foram baseados no trabalho de (OLIVEIRA; SANTOS; RODRIGUES, 2014) [11], utilizando-se uma plataforma de construção de aplicativos e um sensor *bluetooth* de fácil configuração e alcance necessário a este projeto, respectivamente.

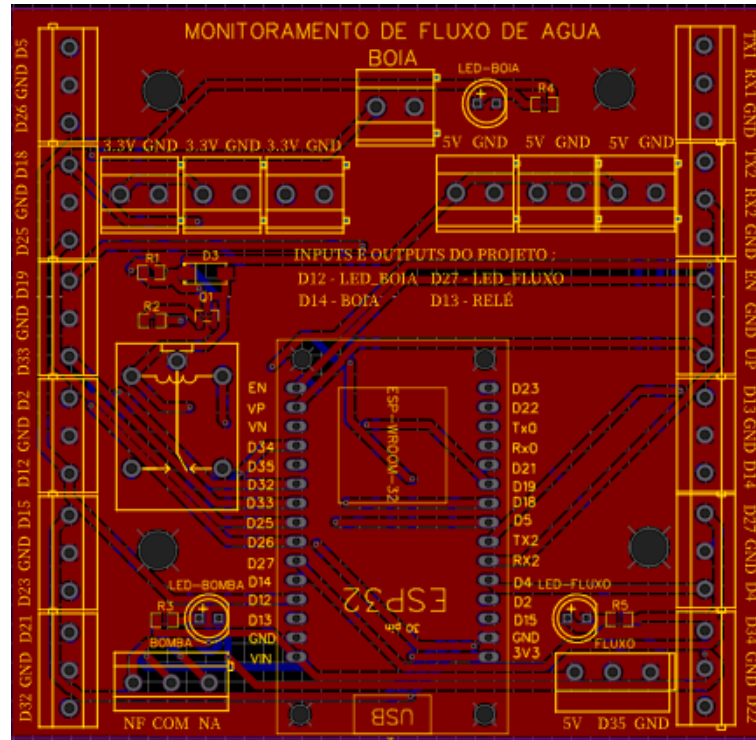


Figura 5: PCB Layout.

3.1 Hardware

Nesta seção serão abordadas as especificações dos componentes de hardware utilizados para desenvolver o projeto proposto nesse trabalho.

3.1.1 Esp32 Wroom Devkit v1

O *esp32* é uma série de microcontroladores de baixo custo e baixo consumo de energia em um chip com *Wi-Fi* integrado e *bluetooth* de modo duplo, como apresentado na Figura 6. A série *esp32* emprega um microprocessador Tensilica Xtensa LX6 em variações *dual-core* e *single-core*, microprocessador Xtensa LX7 *dual-core* ou um microprocessador RISC-V *single-core* e inclui interruptores de antena integrados, RF balun, amplificador de potência, amplificador de recepção de baixo ruído, filtros e módulos de gerenciamento de energia. O *esp32* é criado e desenvolvido pela *Espressif Systems*, uma empresa chinesa com sede em Xangai, e é fabricado pela TSMC usando seu processo de 40 nm. [15] É um sucessor do microcontrolador ESP8266. [16] Neste trabalho foi utilizada uma placa desenvolvimento que consta com as seguintes especificações:

- CPU: Microprocessador LX6 Xtensa dual-core (ou single-core) de 32 bits, operando

DOIT ESP32 DEVKIT V1 PINOUT

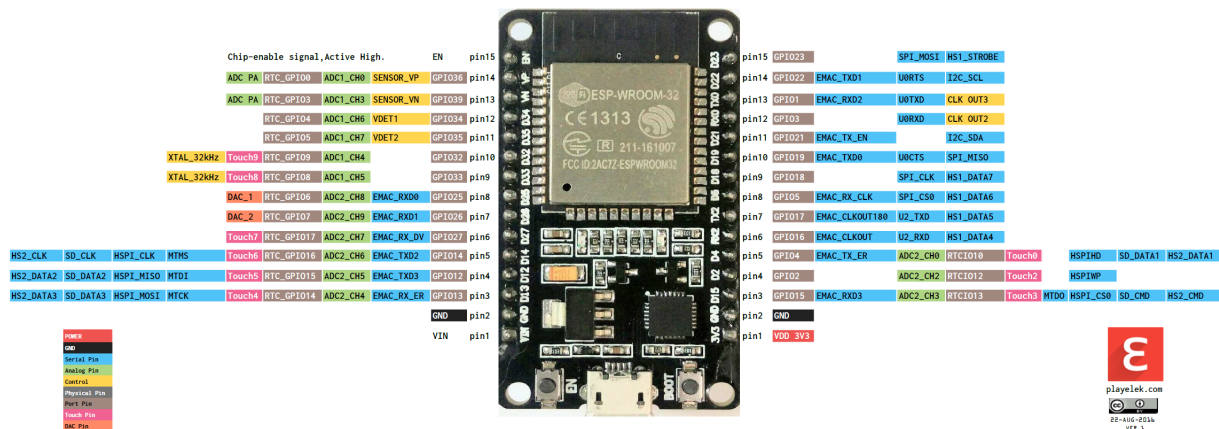


Figura 6: Esp32 Wroom Devkit v1.

a 160 ou 240 MHz e executando até 600 DMIPS Coprocessador de ultra baixa potência (ULP)

- Memória: 520 KiB SRAM
- Conectividade sem fio:
 - Wi-Fi: 802.11 b/g/n
 - Bluetooth: v4.2 BR/EDR e BLE (compartilha o rádio com Wi-Fi)
- Interfaces periféricas:
 - SAR ADC de 12 bits até 18 canais
 - 2 × DACs de 8 bits
 - 10 × sensores de toque (GPIOs de detecção capacitiva)
 - 4 × SPI
 - 2 × interfaces I²S
 - 2 × interfaces I²C
 - 3 × UART

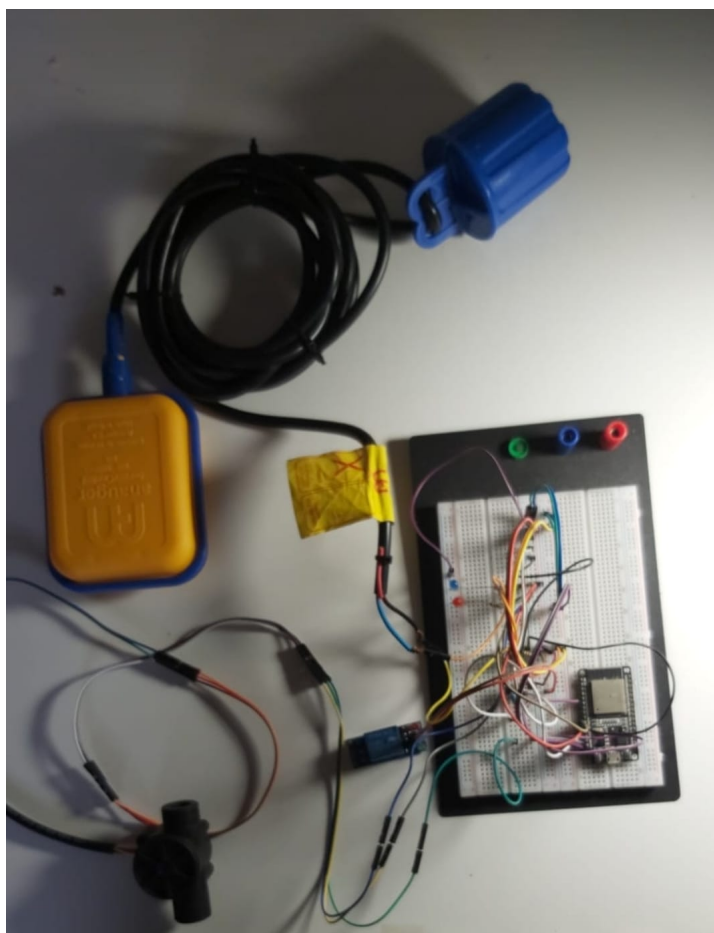


Figura 8: Fotografia da protoboard do projeto.

Um componente que merece destaque: o Microcontrolador. Graças à escolha de utilizá-lo, é possível desenvolver todo o *firmware* (código utilizado no projeto) usando a IDE do *Arduino* (*Arduino Sketch*), tornando o processo muito mais prático. Ademais, a própria plataforma *blynk* que criou módulos de operação e exemplos de códigos das várias funções do app, facilitando a curva de aprendizado e reduzindo o tempo necessário para se habituar com o mesmo.

Abaixo segue a listagem de todo o *hardware* utilizado na *protoboard* e na PCB:

- Esp32 Wroom Devkit v1
- Sensor de Fluxo 1/2 YF-S201b
- ANAUGER Sensor(Bóia)
- Módulo Relé 12V - NA187
- Diodo 1N4007 SMD (M7)

- Transistor BC547 SMD
- 11 conectores KRE de 3 vias
- 7 conectores KRE de 2 vias
- 5 resistores 100Ohm SMD 0804
- 3 LEDS difusos (Vermelho, azul e verde)

3.1.3 Sensor de fluxo

O Sensor de Fluxo de Água Fluxostato 1/2" YF-S201B é uma pequena válvula equipada com um rotor em formato de catavento com um ímã acoplado. Essa estrutura trabalha em par com um sensor de efeito *hall* que envia um sinal *PWM*. Ligado a um sistema de corrente de água, quando o líquido flui através do rotor, ele irá girar e influenciará diretamente na frequência de pulsos emitidos. Cada pulso tem amplitude de aproximadamente 2,25V, e através desses pulsos que são entregues pelo sensor de efeito *hall*, é possível saber qual a vazão de água. [18]

O Sensor de Fluxo de Água Fluxostato 1/2" YF-S201B é largamente utilizado em projetos domésticos de automação residencial, podendo medir o consumo diário, semanal e mensal de uma residência. Pode também ser utilizado para distribuição correta de água entre diferentes caixas d'água ou mesmo atuar como um controlador de nível de água, baseado na entrada e saída de líquido de determinado recipiente.

Esta chave de fluxo apresentada na Figura 10 consegue medir a vazão de água de 1 a 30 litros por minuto e opera com uma pressão máxima de 1,75 Mpa, se mostrando muito eficiente e versátil. Além disso, é desenhado com roscas em suas extremidades para facilitar a conexão com luvas hidráulicas.



Figura 9: Fotografia do Sensor de Fluxo de Água 1/2 YF-S201b.



Figura 10: Sensor de Fluxo de Água - Interno.

3.1.4 Boia de nível

ANAUGER Sensor é um interruptor tipo boia para controle e indicação do nível de água em poços ou reservatórios, com grande precisão e durabilidade. Projetado e fabricado com elementos não tóxicos e matéria-prima de alta qualidade, que proporcionam ao produto um grande desempenho. [19]

- ANAUGER sensor é um interruptor tipo boia para controle e indicação do nível de água em poços ou reservatórios com grande precisão e durabilidade.
- Protege a bomba contra o trabalho sem água (pane seca) ou controla o nível em reservatório evitando o seu transbordamento.
- Liga bomba com reservatório vazio e desliga com reservatório cheio.
- Liga bomba quando há água e desliga na falta de água.



Figura 11: Fotografia da bóia com contrapeso

3.1.4.1 Funcionamento da bóia

A bóia atua como um relé comum, utilizados normalmente como uma chave eletrônica. Como ilustrado na Figura 12, temos o fio marrom(NF), preto(NA) e azul (comum). [1]

Em termos práticos utiliza-se o preto e o azul, onde teremos o contato fechado entre os mesmos toda a vez que a bóia estiver na posição vertical para baixo e normalmente aberto quando ela estiver na posição vertical para cima como mostrado na Figura 13.

Em resumo teremos a chave ligada quando o nível da água estiver muito baixo de maneira que a bóia desça até a posição vertical, e quando a bóia estiver na posição vertical superior, devido ao nível da água estar alto a chave desliga.

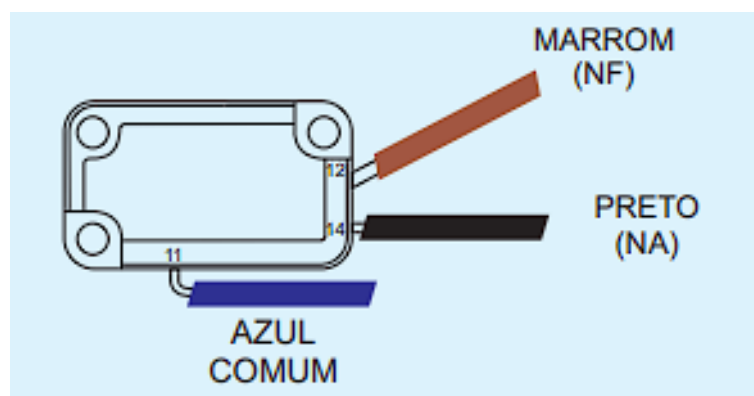


Figura 12: Relé interno à bóia [1].

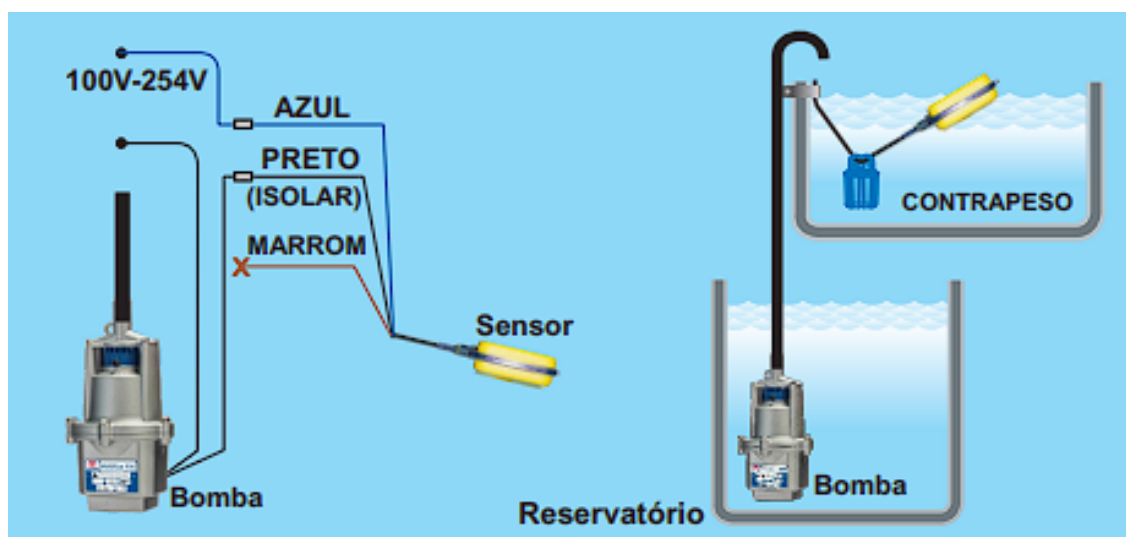


Figura 13: Esquemas de ligação da bóia [1]

3.1.5 Relé

O módulo de relé de 12V de 1 canal é uma placa de interface de relé de 12V de 1 canal com terminal de parafuso que pode ser controlado diretamente por uma ampla gama de microcontroladores, como Arduino, *AVR*, *PIC*, *ARM* etc. A placa possui um relé de alta qualidade, que suporta no máximo 10A - 125VAC ou 10A- 250V AC. O relé tem três conexões - Comum - C, Normalmente Aberto-NA, Normalmente Fechado-NF trazidos para terminais de parafuso de 3 pinos, o que facilita fazer e remover conexões. A placa possui uma indicação de energia e um *LED* de status do relé para facilitar a depuração. A entrada de energia e os sinais de controle do relé são levados aos pinos principais na placa. Portanto, a placa pode facilmente interagir com nossas placas de desenvolvimento usando nossos fios de jumper. [2]

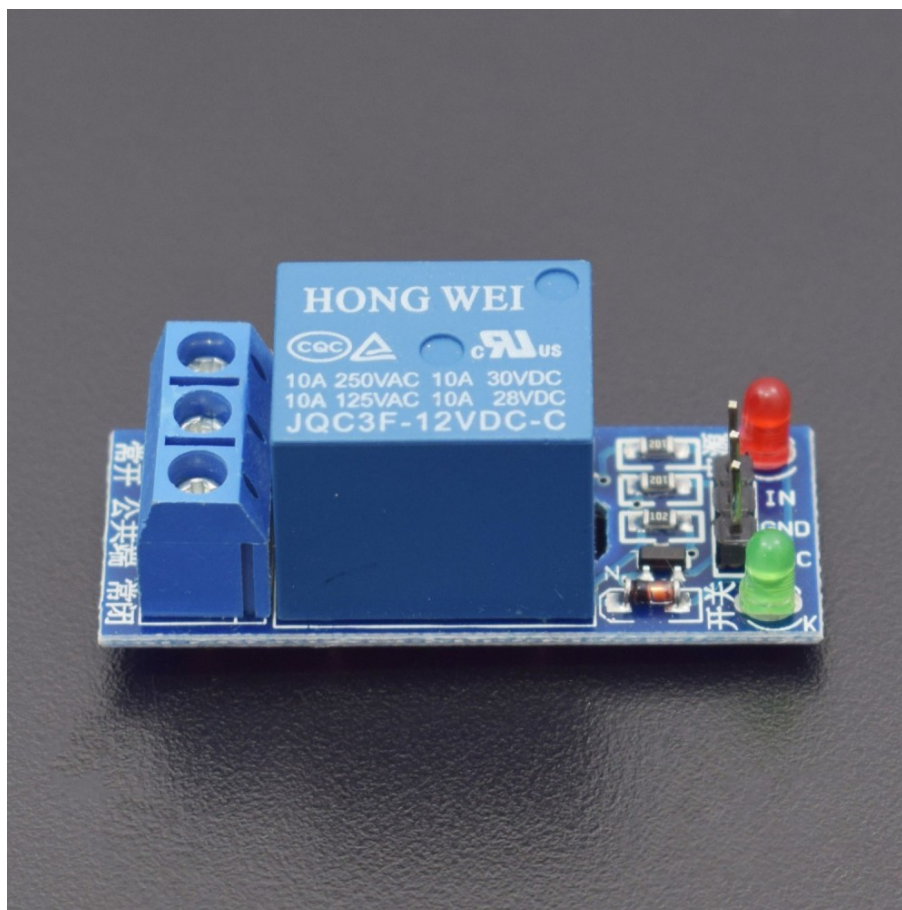


Figura 14: Esquemas de ligação da bóia [2]

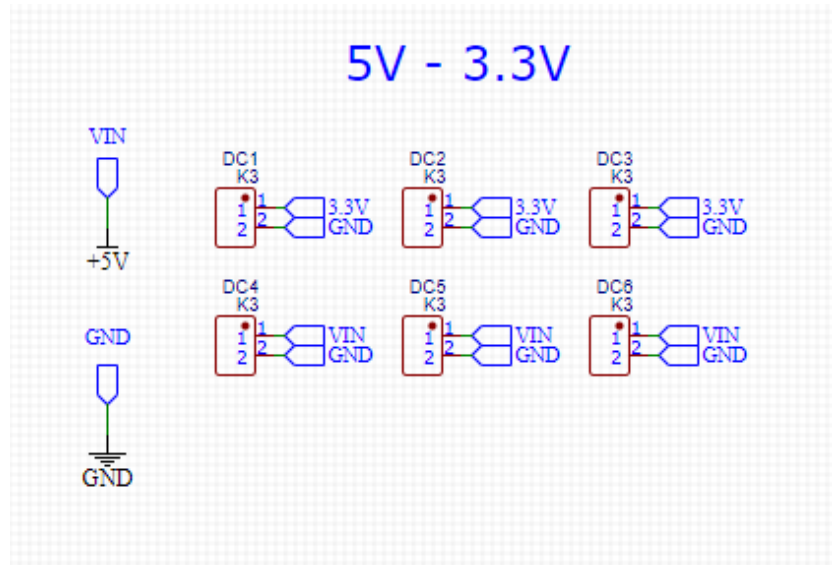


Figura 17: Esquema elétrico - Tensões.

O bloco do sensor de fluxo é bem simples, como foi explicado na subsecção 3.1.3. Quando o *PWM* é gerado pelo sensor devido à passagem de água no seu rotor, o *esp32* o recebe pelo pino digital 35 do ESP32, que pelo seu código manda um sinal de nível lógico alto para o pino digital 27 a cada 1 segundo, acendendo o *LED3* (azul) ilustrado na Figura 18 do esquema e também pode ser visto na PCB da Figura 32.

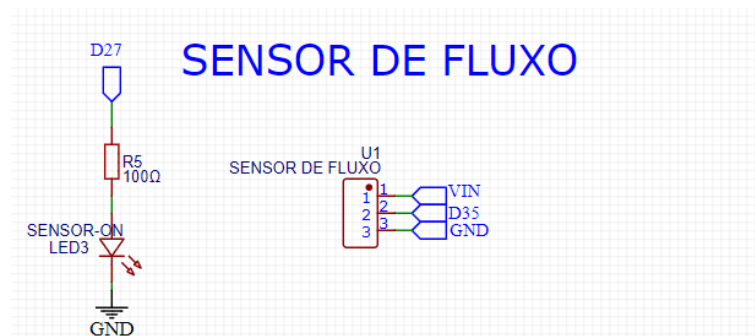


Figura 18: Esquema elétrico - Sensor de fluxo

No próximo bloco na bóia temos um resistor de *PULL-UP* que mantém o nível lógico alto na *label* “BOIA” no digital 14, como já foi explicado na subsecção 3.1.4. Quando a bóia vai descendo por conta do nível baixo na caixa d’água, ela fecha um curto nas *labels* “BOIA” (fio azul da bóia), “BOIA” (fio preto da bóia). O digital 14 vai para o nível lógico baixo, acionando a programação para liberar um sinal de nível lógico alto para o digital 12, acendendo a *LED2* indicando o nível baixo de água na caixa. Podemos ver isso no esquema representado na Figura 19 e também veremos a *LED* vermelha acendendo na PCB da Figura 30.

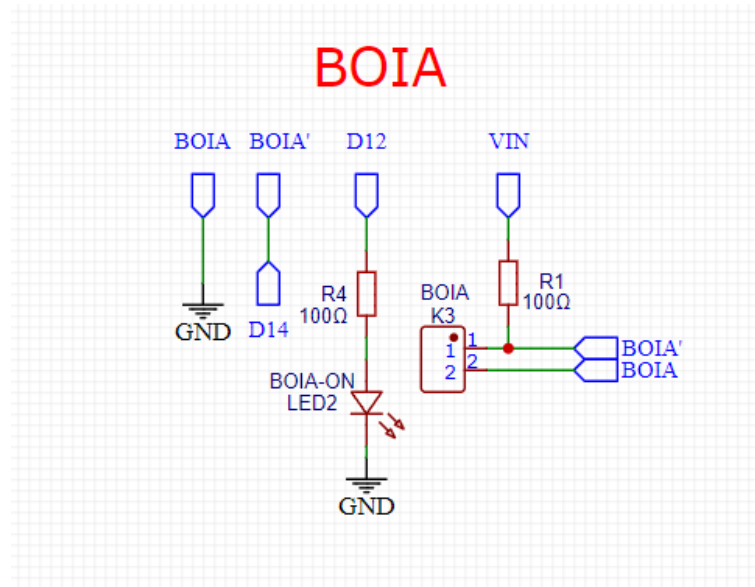


Figura 19: Esquema elétrico - Bóia.

O bloco do relé é altamente dependente das duas últimas seções anteriores, ou seja, ele só é acionado quando o nível de água estiver baixo e tivermos água na rua. Portanto quando a água passar pelo sensor de fluxo e o nível de água dentro da caixa d'água estiver baixo. Com esses critérios satisfeitos, a programação enviará um sinal de nível lógico alto para o digital 13, que por sua vez vai saturar a base do transistor Q1 da Figura 20 chaveando o ânodo do diodo D3 e o cátodo da *LED1* no *GND*, acendendo a *LED* verde da Figura 34 e acionando o relé mudando o estado do conector KRE, chaveando o NF para aberto e o NA para fechado, ligando a bomba.

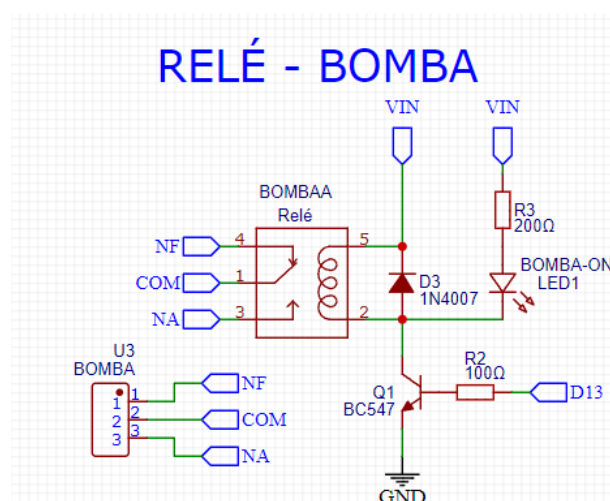


Figura 20: Esquema elétrico - Relé.

3.1.7 Estimativa de Orçamento

Tendo todos os componentes definidos, é possível estimar o custo da implementação na Tabela 1. Vale comentar que este é apenas um caminho tomado, tendo múltiplas formas de implementar uma solução que alcance os resultados desejados.

Tabela 1: Preços estimados para componentes descritos na solução.

Item	Preço (US\$)	Preço (R\$)
Prototipagem da PCB	-	138,26 [17]
Esp32 Wroom Devkit v1	-	65,00 [20]
Sensor de Fluxo 1/2 YF-S201b	-	36,00 [21]
ANAUER Senso	-	67,00 [22]
Módulo Relé 12V - NA187	-	29,90 [23]
Diodo 1N4007 SMD (M7)	-	0,20 [24]
Transistor BC547 SMD	-	0,30 [25]
11 conectores KRE de 3 vias	-	20,90 [26]
7 conectores KRE de 2 vias	-	9,10 [27]
5 resistores 100Ω SMD0804	-	2,50 [28]
3 LEDs Difuso	-	0,75 [29]
Total		369,91

3.2 Software

3.2.1 Linguagem Arduino Sketch

A IDE utilizada no projeto foi a do *Arduino*, que é um *framework* baseado em uma versão simplificada na linguagem C e C++, o que torna mais fácil e adaptável as sintaxes e as nomenclaturas dos comandos da linguagem Arduino, facilitando a vida dos iniciantes, estudantes e entusiastas.

A IDE compila o código que escrevemos, e transforma em linguagem *assembly*, linguagem utilizada pelos Chips Microcontrolador Atmel, nas placas *Arduino*.

A linguagem de programação do *Arduino* não está limitada apenas à linguagem C e C++. Ele permite outros tipos de linguagens, como *Python*, sendo necessário a instalação de uma biblioteca na *Arduino* IDE. Existem duas bibliotecas bastante conhecidas, o *pyFirmata* e o *pySerial* disponíveis para construir uma interface entre o *Shell* do desenvolvedor *Python* e a IDE *Arduino*. [12]

3.2.2 Ferramenta *blynk*

O *blynk* foi desenvolvido para ser utilizado em projetos *Iot* (*Internet of Things* ou Internet das coisas, em Português), que é um termo utilizado para descrever a forma como objetos do mundo real permanecem conectados em rede e podem ser acessados através da Internet. [9].

A principal característica do *blynk* é permitir que a comunicação com uma plataforma microcontrolada / embarcada possa ser realizada sabendo o mínimo de programação e até mesmo sem criar uma linha de código. Ele permite que plataformas sejam controladas remotamente, de forma que dados de sensores e módulos possam ser obtidos e exibidos no aplicativo que fica instalado no dispositivo móvel. Permite também que cargas sejam acionadas, além de muitas outras funcionalidades que a ferramenta disponibiliza. [3]

Um dos pontos cruciais do *blynk* é a possibilidade de controlar a plataforma de qualquer lugar do mundo através do aplicativo instalado no dispositivo móvel. Para isto, basta que a plataforma esteja configurada e conectada ao servidor Blynk através da internet e que o aplicativo no dispositivo móvel também possua conexão com a internet. Uma pequena ilustração disto está ilustrada na Figura 21 .

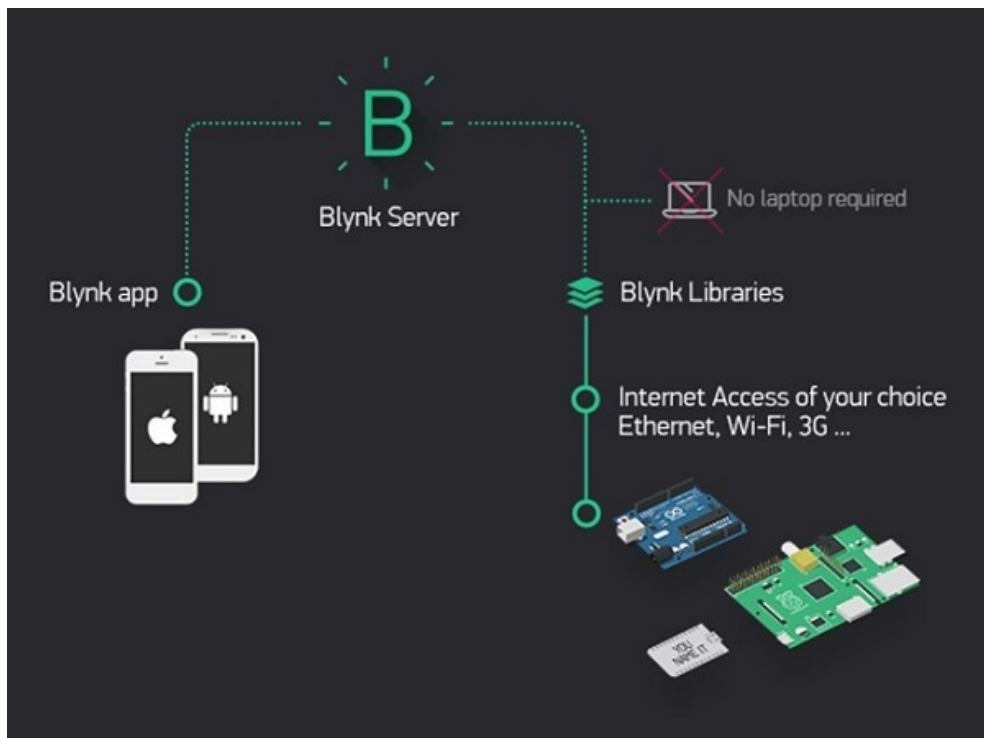


Figura 21: Comunicação entre o dispositivo móvel, servidor do *blynk* e o *esp32* [3].

3.2.3 Configuração da plataforma *blynk*

O app *blynk* é que vai estabelecer a comunicação entre módulos *esp32* e o celular. Inicialmente na configuração do *software* do *blynk* teremos a criação do projeto com o nome de “Projeto monitoramento de fluxo de água teste”. Na criação do mesmo teremos o recebimento do “BLYNK_TEMPLATE_ID” e o “BLYNK_AUTH_TOKEN”, que estão no código no repositório do *github* [13], e são valores que o próprio aplicativo do *blynk* nos fornece ao criar o projeto. Valores esses que são essenciais para a comunicação do *esp32* com o servidor do *blynk* com o projeto ilustrado na Figura 22.

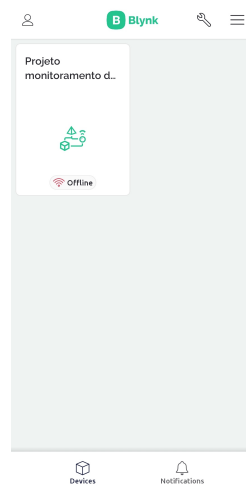


Figura 22: Criação do projeto na plataforma *blynk*.

3.2.4 Interface e *widgets* do *blynk*

Após a criação do projeto, teremos a criação em si da interface que estabelecerá a comunicação com todo o *hardware*, recebendo valores de cada sensor e acionando os *LED* virtuais e utilizando variáveis virtuais para notificar o seu funcionamento com mensagem através de *widgets* do próprio *blynk*. A interface inicial foi dividida em duas abas uma com o nome “Gerenciador” e outra com o nome “Painel de eventos”. No gerenciador podemos visualizar as *LED* virtuais e o indicador de vazão de água, enquanto no painel de eventos serão marcados os dias e horas que ocorrerão os eventos de nível baixo de água, caixa cheia, tem água da rua e não tem água na rua.

Utilizando o menu de *widgets* do *blynk* ilustrado na Figura 23, foram utilizados 3 *LED* virtuais, nas quais cada uma irá indicar o acionamento de cada sensor em específico, assim como foram utilizadas na seção 3.1.6 do esquema elétrico as LEDs físicas: vermelho

para a boia, azul para o sensor de fluxo e a verde para o relé. As LEDs virtuais seguiram este mesmo raciocínio.

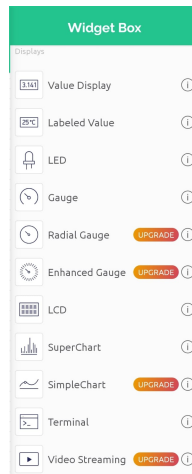


Figura 23: Caixa de Widget do *blynk*.

A primeira *LED* virtual será a da boia, ela terá o nome “NÍVEL DE ÁGUA BAIXO!” e foi associada à variável virtual 2 (V2), como mostra a Figura 24. Com essa variável podemos controlar essa *LED* pela IDE do Arduino.

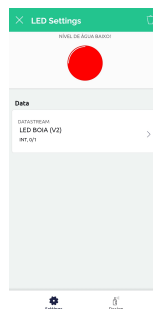


Figura 24: *LED* virtual - Boia

Utilizando novamente o menu de widgets do *blynk* da Figura 23, foi utilizado, mais uma *LED* virtual, na qual essa em específico, com o nome “TEM ÁGUA NA RUA” foi associada à variável virtual 1 (V1). Ela será a *LED* responsável pelo fluxo de água. Podemos ver isto na Figura 25.

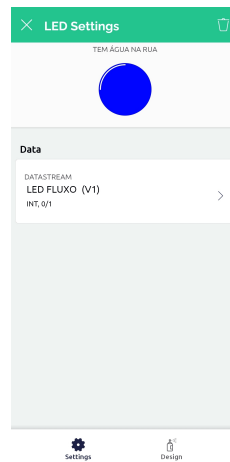


Figura 25: *LED* virtual - Sensor de fluxo.

Após a passagem de água pelo sensor de fluxo, a programação chama a função *fluxo_agua()*, que calcular a vazão de água que está passando pelo sensor, função esta que calcula a vazão de água pela quantidade de pulsos recebidos pelos *esp32* e seu valor armazenado na variável virtual 3 (V3), como apresentado na Figura 26.

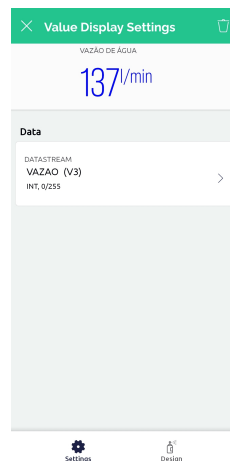


Figura 26: Display da vazão de água.

Por fim foi utilizado mais uma *LED*, associada à variável virtual 2 (V2). Como ilustrado na Figura 27, ela acende sempre quando a variável “contaPulso” é maior que 1 e quando a variável “BOIA” está em nível lógico baixo (*LOW*), entrando no *if* do código ele aciona as *LED* físicas e virtuais do fluxo e do relé e ligando também o relé. Por fim ele deixa piscando a *LED* da bóia a cada um segundo para indicar o enchimento da caixa d’água. Podemos ver tudo isso funcionando na Figura 35.

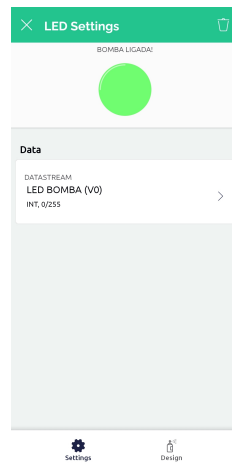


Figura 27: *LED* virtual do relé.

3.2.5 Painele de eventos

O painel de eventos da Figura 28 é onde temos um *log* que registra os dias e horas em que ocorreram os eventos relacionados ao sistema. Ele utiliza somente *displays* para expressar os valores que são recebidos de data e hora, sendo a variável virtual (V4) para indicar o dia, as variáveis V5,V10,V11 e V12 foram utilizadas para as mensagens “Caixa está vazia!”, “Caixa encheu!”, “Não tem mais água na rua!”e “Chegou água da rua”, respectivamente podendo ser vistas na Figura 29 e as variáveis V6, V7, V8 e V9 para expressar a hora de cada um desses eventos também.

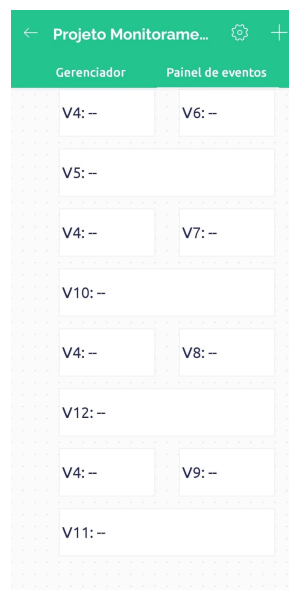
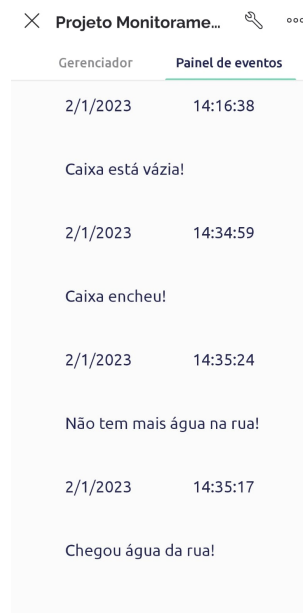


Figura 28: Painele de eventos com as variáveis.



The screenshot shows the Blynk mobile application interface for a project named 'Projeto Monitorame...'. At the top, there is a search icon and a menu icon. Below the title bar, there are two tabs: 'Gerenciador' and 'Painel de eventos', with the latter being the active tab. The main area displays a list of events, each with a date, time, and a message.

2/1/2023	14:16:38	
		Caixa está vazia!
2/1/2023	14:34:59	
		Caixa encheu!
2/1/2023	14:35:24	
		Não tem mais água na rua!
2/1/2023	14:35:17	
		Chegou água da rua!

Figura 29: Painei de eventos do *blynk*.

3.3 Funcionamento do projeto

O projeto prático com a PCB e todo o sistema funcionando em sua totalidade com o aplicativo *blynk* está no vídeo do youtube [30] mostrado no anexo desse arquivo, semelhantemente ao código que está no diretório do *github* [13].

Inicialmente o projeto é composto por dois blocos, sendo eles *software* e *hardware*, o *hardware* é quem permite todas as interações do mundo real com o digital. O primeiro dispositivo de entrada de sinal será a bóia elétrica, que vai ser conectada no conector KRE de duas vias. Como podemos ver na Figura 7 é possível visualizar o conector KRE de duas vias com o nome conector da bóia, onde são conectados os fios azul e preto da bóia elétrica da Figura 11. Ela fecha e abre o contato toda vez que o nível da bóia desce e sobe respectivamente. Não existe uma polaridade no conector da boia , pois são contatos secos. Eles somente abre e fecha como um relé comum. Toda vez que o nível de água está baixo, a bóia fecha o contato. Sendo assim o *esp32* acende a *led* virtual do aplicativo na Figura 31 e a *led* física da PCB na Figura 30.

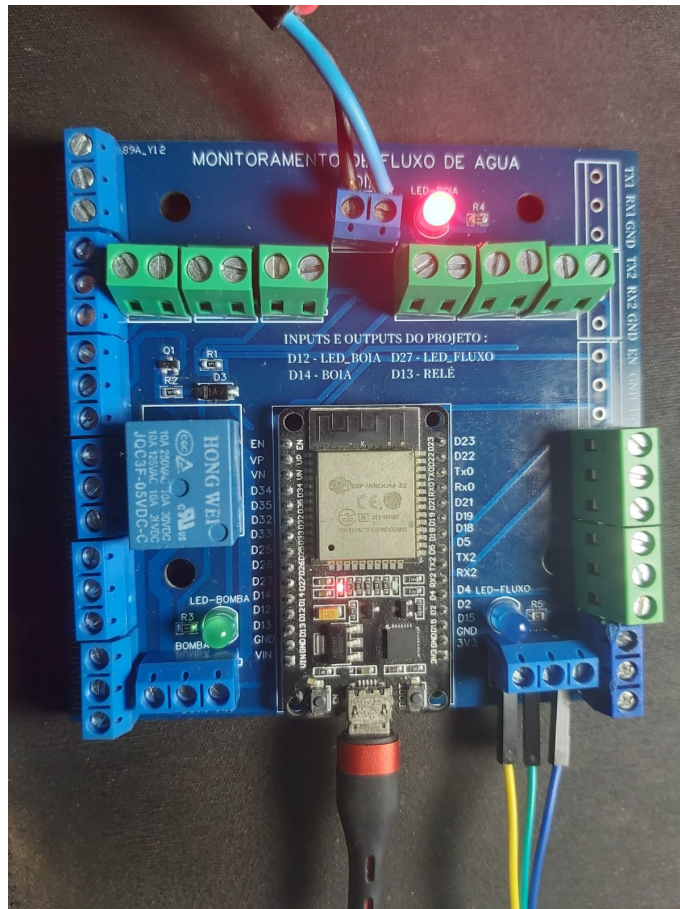


Figura 30: Led física da boia.



Figura 31: Gerenciador dos Widgets.

A outra aba do aplicativo que também é atualizada ao mesmo tempo que a das *LEDs* é a do painel de eventos da Figura 29. Toda vez que a *led* da boia acende, ele registra a hora que foi acessa, indicando o horário que a caixa ficou vazia. Quando o nível da água sobe, a *led* virtual e física apagam, registrando no painel de eventos o horário que a caixa encheu. Toda a sua configuração pode ser vista na Seção 3.2.5.

Em seguida temos o sensor de fluxo, que é outro sinal de entrada. Podemos visualizar na Figura 9 três fios : vermelho(VCC), preto(GND) e amarelo(Sinal), os quais vão para o conector KRE de três vias. Seu funcionamento é bem simples; mandar um sinal de *PWM* pelo fio amarelo toda a vez que tiver a passagem de água no seu rotor interno que gera o *PWM* por efeito hall, sinalizando para o *esp32* que existe fluxo de água na rua.

Quando isto acontece, a *led* física da PCB e a virtual do aplicativo acendem e ficam piscando a cada um segundo como podemos ver na Figura 33 e na Figura 32. Ao mesmo tempo que as leds piscam, é registrado também o horário deste evento no painel de eventos, registrando a hora que acaba e chega a água.

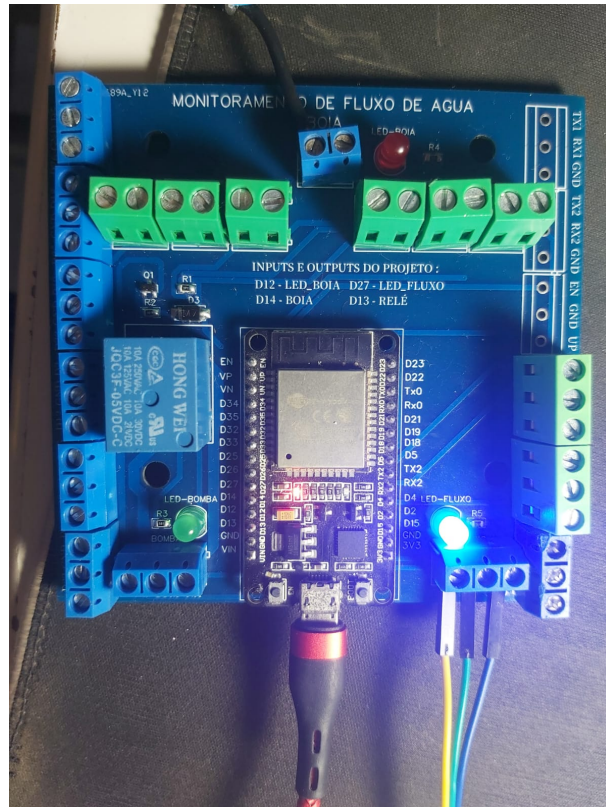


Figura 32: Led física do sensor de fluxo.



Figura 33: Led virtual do fluxo.

Por fim, quando temos a boia com o contato fechado devido ao nível baixo de água na caixa e o fluxo de água no sensor mandando o *PWM* devido à presença de água na rua, o *esp32* aciona o relé que fecha o contato e liga a bomba como mostra a Figura 34. Podemos ver na Figura 35 a tela do aplicativo do *blink*.

Até o enchimento da caixa d'água temos o acionamento da bomba. No decorrer que a bóia elétrica vai subindo devido ao aumento do nível de água, ela chega ao nível máximo, a bóia abre o contato e desliga todo o sistema.

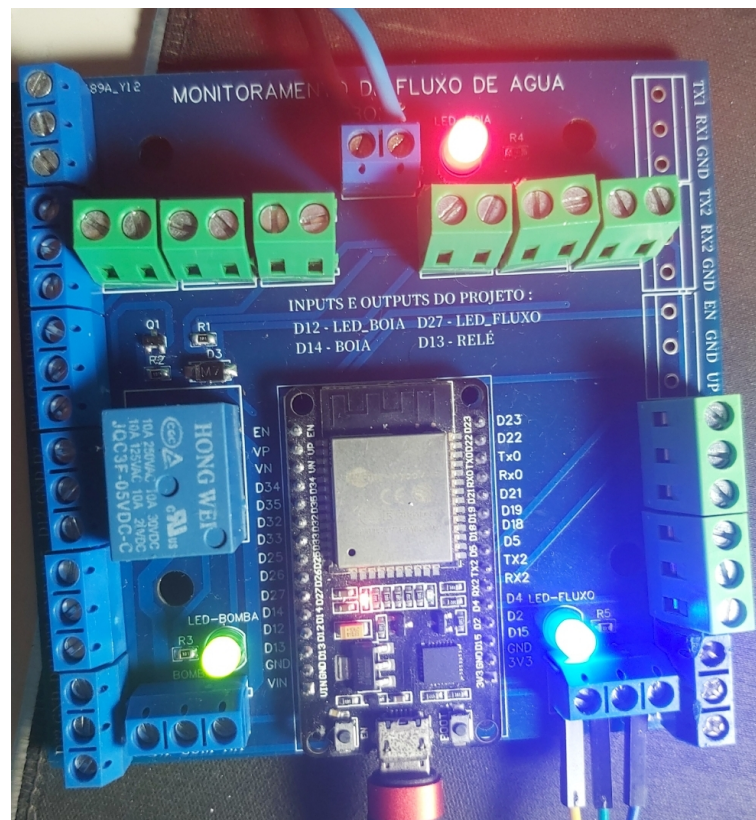


Figura 34: Led física do relé.

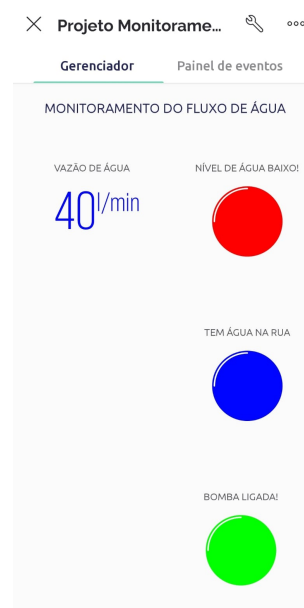


Figura 35: Led virtual do relé.

4 TESTES

Como foi mencionado na secção 3.3, podemos ver o funcionamento do projeto na prática em uma residência, testando a PCB e juntamente com o aplicativo acessando o link do youtube [30], que mostra toda a bateria de testes práticos desde os dias que não havia água até ter água na rua.

4.1 Local dos testes

Para confirmar a praticidade com excelência do projeto, ele foi implementado em uma situação real na qual o cenário em questão é uma residência que tem deficiência de água em sua região.

O local escolhido em específico para a fixação da placa dentro desta residência foi selecionado pelo simples fato de estar próximo a todos os elementos que fazem interação com a PCB, como podemos ver na Figura 36. A placa foi fixada utilizando pregos convencionais de alguns fixa-fio atravessam a PCB nos furos que assim foram projetados com esta finalidade.

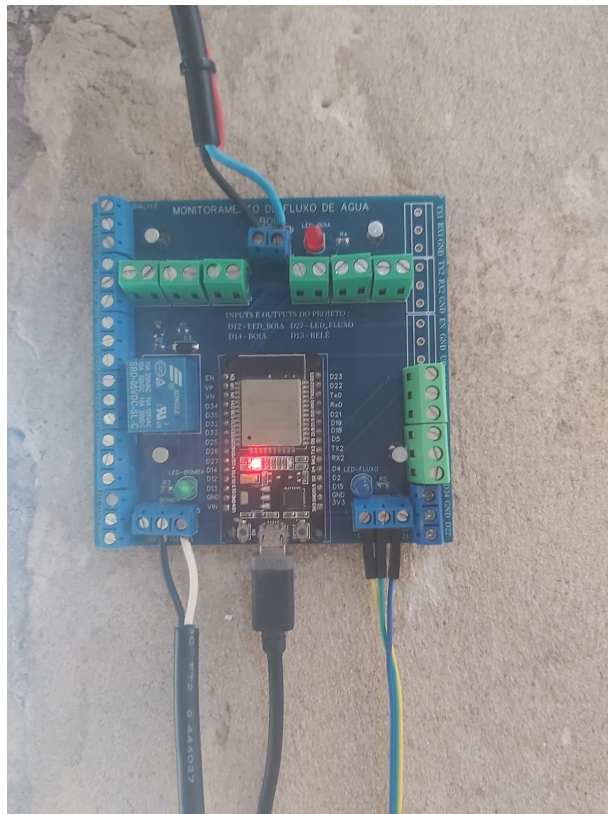


Figura 36: PCB fixada no local dos testes.

4.1.1 Instalação da boia

A instalação da boia começa na PCB onde podemos ver na Figura 37 dois fios (azul e preto), conectados na PCB na sua parte superior. Esses fios levam até a caixa d'água. Quando os fios chegam na caixa, eles são fixados no cano com uma fita isolante de alta qualidade da 3M, cano este que leva a água da bomba para encher a caixa.

A fixação com ele fixado no cano está na Figura 38 apresentada. Numa altura específica para que o contra-peso permaneça a uma altura aproximadamente de 50% do nível de água na caixa, sendo assim possível a bóia funcionar sempre quando o nível de água descer abaixo de 50%.



Figura 37: Instalação do fio da bóia na PCB.



Figura 38: Fio da bóia dentro da caixa d'água.

Como podemos ver a Figura 39, a bóia encontra-se flutuando sobre a água bem próxima do limite de nível baixo. No momento ela fica nesta posição até o nível descer totalmente e fechar o contato do relé interno informando a PCB de nível baixo de água.



Figura 39: Bóia dentro da caixa d'água.

Em outro instante do mesmo dia foi verificado novamente o nível de água dentro da caixa d'água e foi possível constatar o nível abaixo de 50%, acionando a *LED* física e virtual da bóia da PCB na Figura 40 e no aplicativo na Figura 31.

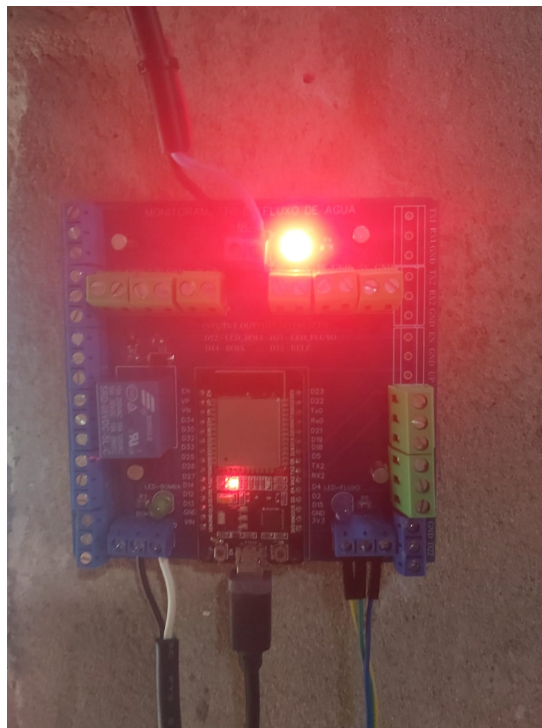


Figura 40: *LED* indicadora de nível baixo na caixa.

Sendo assim após ocorrer todos esses eventos o painel de eventos registra data e

hora em que ocorreu tudo isto, conforme ilustrado na Figura 41.

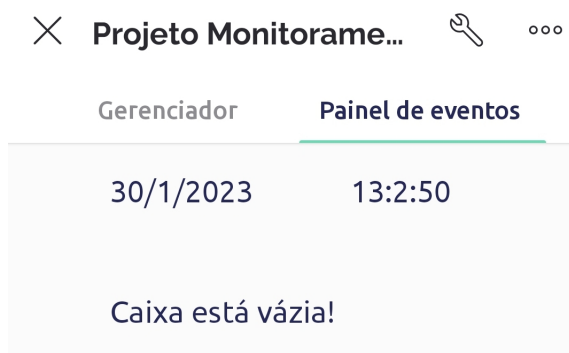


Figura 41: Hora que o nível baixo de água ficou baixo no painel de eventos.

4.1.2 Instalação do sensor de fluxo

O sensor de fluxo depende diretamente de água para o seu funcionamento. Seu local de instalação é após o hidrômetro e antes da bomba d'água. Sendo assim ele foi colocado na tubulação deste modo como mostra a Figura 42 .

Inicialmente foi colocado na encanação em um dia que não tinha água da rua para não dificultar a sua instalação. Foram utilizados duas luvas de cano de 25mm um lado rosqueado e outro liso para poder dar o encaixe correto na encanação. Após colocado no lugar, a *LED* da placa permanece apagada como mostra a Figura 36, pois não tinha água da rua neste dia, consequentemente não havia fluxo.



Figura 42: Sensor de fluxo instalado na encanação.

Em outro dia da semana (31/01/2023) com água na rua, foi verificada a existência de fluxo de água. Sendo assim a passagem desse fluxo foi conferida pelo sensor ao passar água pelo seu rotor. Ao fazer isso, a *LED* azul começa a piscar a cada segundo, sendo ela física ou a virtual ambas se comportando do mesmo jeito quando temos o fluxo de água na passagem do rotor do sensor.

Na PCB podemos ver na Figura 43 o acendimento da *LED* física e virtual piscando simultaneamente no aplicativo. Por fim temos o painel de eventos, que vai registrar o dia e a hora em que ocorreu a passagem de água no rotor do sensor de fluxo. Observando a Figura 44 podemos ver o dia e a hora em que isto ocorreu.

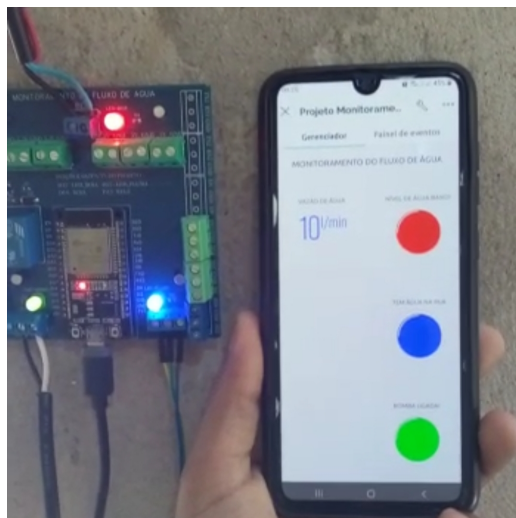


Figura 43: Sistema funcionando na PCB e no aplicativo.

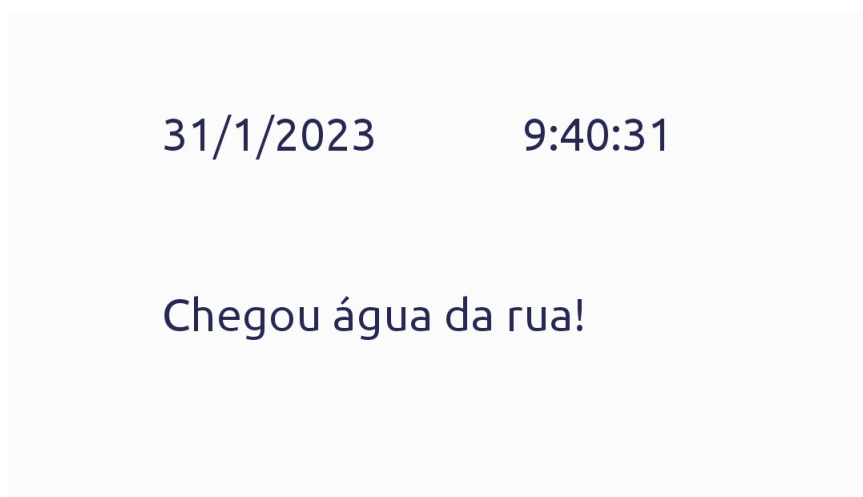


Figura 44: Data e hora em que chegou água na rua.

4.1.3 Instalação do relé com a bomba

A instalação do relé com a bomba foi feita com um fio paralelo preto para o comum e branco para o normalmente aberto. Os dois trabalham como uma chave, ou seja, foi feito um corte no fio da bomba para que a mesma seja controlada somente pelo relé. Feito esse corte, o fio preto foi para o AC da rede elétrica e o branco foi direto para a bomba deixando o fio preto e branco fechados toda vez que o relé for acionado e aberto toda vez que o relé estiver desligado. Toda esta conexão na PCB pode ser vista na Figura 45.

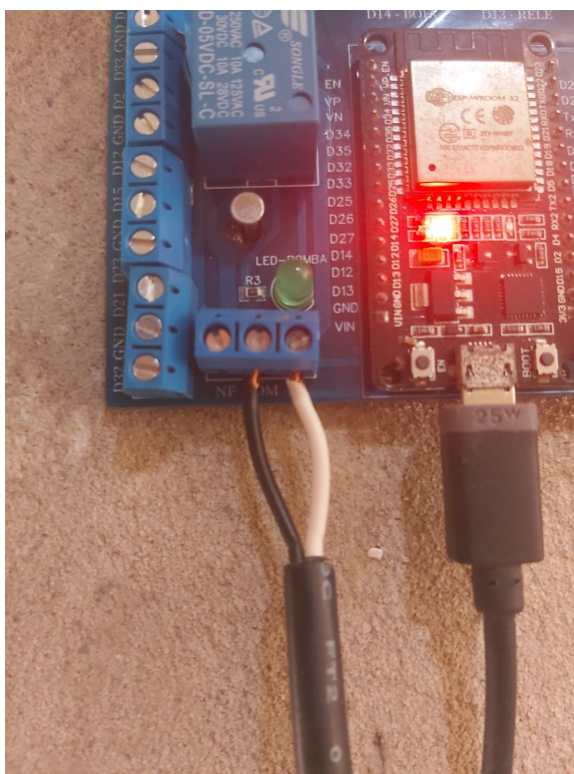


Figura 45: Ligação elétrica do AC da bomba com o relé.

Seu acionamento foi feito quando o nível de água estava baixo e tivemos mais de um pulso no sensor de fluxo como mostra a Figura 43. Quando estes dois critérios foram satisfeitos, o relé acionou e ligou a bomba, dando início ao processo de enchimento da caixa d'água, como podemos ver na Figura 46.

O processo do enchimento da caixa d'água se dá pelos dois critérios acima citados, ou seja, a bomba só desliga quando o nível de água estiver alto, ou a água da rua não estiver mais nas encanações da residência.

Na Figura 47 podemos visualizar o desligamento da bomba, pois a caixa já encheu e o nível da boia subiu, abrindo o contato e desligando o sistema por completo.



Figura 46: Enchimento da caixa d'água.



Figura 47: Boia desligando o sistema devido ao nível alto de água na caixa.

5 DIFICULDADES ENCONTRADAS

As principais dificuldades encontradas ao longo do desenvolvimento deste trabalho foram:

- Informações e aplicações utilizando a plataforma do *blynk* foram bastante complicadas, pois a interação do software na IDE do *Arduino* com a interface do aplicativo e o servidor do *blynk* exige um bom domínio de variáveis virtuais da biblioteca do *blynk* e saber manipular as mesmas leva um tempo. Em contrapartida o código foi bastante reduzido devido à sua plataforma, pois desenvolver uma aplicação para fazer tal gerenciamento seria bem mais complexo.
- Acesso a componentes de *hardware*. Todo o material de *hardware* foi comprado na recicomp, com excessão da placa que foi feita pela jlcpb. Toda a soldagem foi feita manualmente, utilizando a solda cobix de 1mm. A soldagem teve umas dificuldades devido a alguns pads pelo tamanho, mas tudo foi resolvido com um bom ferro de solda da hikari.
- Na prototipagem da PCB foi complexo fazer as trilhas no autoroute do *EasyEDA(Standard)*, pois uma placa com tantas trilhas teve alguns erros de DRC. Como a mesma foi dimensionada para 10cm por 10cm para diminuir o custo de projeto, para resolver este problema de trilhas em excesso foi feita uma *copper area*(máscara de solda).
- Testes práticos. A instalação da PCB e de cada componente foi um desafio que exigiu conhecimentos de pressão de água, encanação e outros que deram um trabalhos a mais. Dando destaque ao sensor de fluxo, cuja instalação na encanação da residência exigiu duas luvas de 25mm com metade rosqueado e o outro lado não rosqueado, uma peça mais difícil de se instalar.

6 CONCLUSÃO E TRABALHOS FUTUROS

A solução proposta consistiu na criação de uma PCB que receberia as informações dos sensores e envia para um aplicativo via *wireless* dando facilidade e automatizando processos repetitivos, que são necessários para o uso da água na residência.

Pelos resultados obtidos, é possível concluir que o objetivo deste trabalho foi atingido, já que o sistema desenvolvido satisfaz todos os objetivos propostos, sem apresentar falhas notáveis em seus comportamentos. Como já foi mencionado na seção 3.3, foi gravado um vídeo de demonstração da PCB com o aplicativo do *blynk* funcionando numa situação real [30].

Foi possível testar exaustivamente o sistema, em torno de uma semana na residência escolhida, obtendo um resultado bastante satisfatório e garantido sua estabilidade nesta localidade.

Além disso, outras atividades podem ser mapeadas como trabalhos futuros, entre elas:

- Testar projeto com mais casos/configurações: Em instalações durante os testes de *wireless*, houve ocasiões pontuais em que a conexão não foi estabelecida com sucesso. Um fator muito importante é a distância da placa com o roteador, pois em uma longa distância teria uma instabilidade na conexão ou mesmo nem seria possível se conectar com a rede de internet.
- Desenvolver uma aplicação específica: desenvolver um *software* para ser independente da plataforma do *blynk*, pois saímos de uma aplicação de desenvolvimento e teste para um *software* focado no propósito principal do projeto, corrigindo alguns erros de conexão e otimizando memória do dispositivo do *esp32*.
- A adição de mais sensores : seria uma boa opção colocar mais sensores para poder estimar a quantidade de água dentro da caixa d'água, para visualizar uma variação abrupta de pressão, possível vazamento nos canos, ter uma noção do custo mensal de água gasto, baseado nos valores cobrados da empresa de distribuição de água e também a adição de uma válvula solenóide que possibilita a abertura e fechamento do registro da água da rua sempre que há uma tensão específica entre seus terminais.

[31]

REFERÊNCIAS

- [1] SENSOR, A. *Bóia elétrica manual*. Available at: <https://anauger.com.br/wp-content/uploads/2019/12/SITE_MANUAL_49020-CAIXA_SENSOR_1_5M_CX.-SENSOR_PORTUGU%C3%8AS_17-MAIO-2019.pdf>. Accessed in: 29/01/2023.
- [2] NA187. *Módulo Relé*. Available at: <<https://rees52.com/12v-1channel-relay-module-na187-1948-na187-general-purpose-relay-module/>>.
- [3] OLIVEIRA, E. *Conhecendo o blynk*. Available at: <<https://blogmasterwalkershop.com.br/blynk/conhecendo-o-blynk>>.
- [4] PENA, A. F. R. *Distribuição da água no Brasil*. 2000. Available at: <<https://brasilecola.uol.com.br/geografia/distribuicao-agua-no-brasil.htm>>. Accessed in: 07/01/2023.
- [5] SARDINHA, R. S. V. *Água*. 2019. Available at: <<https://brasilecola.uol.com.br/geografia/agua.htm>>. Accessed in: 07/01/2023.
- [6] PENA”, P. M. R. A. *Gráfico distribuição de água no brasil*. Available at: <<https://brasilecola.uol.com.br/geografia/distribuicao-agua-no-brasil.htm>>.
- [7] BRKAMBIENTAL. *Saneamento básico no Brasil: conheça os números das regiões do país*. 2020. Available at: <<https://blog.brkambiental.com.br/saneamento-basico-no-brasil/>>. Accessed in: 07/01/2023.
- [8] COMPESA. *Calendário de abastecimento compesa*. Available at: <<https://servicos.compesa.com.br/calendario-de-abastecimento-da-compesa/>>. Accessed in: 29/01/2023.
- [9] SERGIO, L. *What is Internet of Things (IoT)? 139+1 definitions since 1999*. Available at: <<https://www.rtsrl.eu/blog/what-is-internet-of-things-iot/>>. Accessed in: 08/01/2023.
- [10] DORNELAS. Monitoramento de consumo doméstico de Água utilizando uma meta-plataforma de iot. *Revista de Engenharia e Pesquisa Aplicada*, n. 2.

- [11] OLIVEIRA H. R. I.; SANTOS, R. B. C. Desenvolvimento de um aplicativo android para monitoramento microcontrolado do nível de um reservatório de água residencial em tempo real. *Conferência de Estudo em Engenharia Elétrica*, Abr 2014.
- [12] ARDUINO. *About*. Available at: <<https://www.arduino.cc/>>. Accessed in: 22/01/2023.
- [13] PEDRO, D. *Projeto-GERENCIADOR-DE-FLUXO*. Available at: <<https://github.com/Singular123456/TCC-GERENCIADOR-DE-FLUXO-DE-GUA-ESP32.git>>.
- [14] EASYEDA. *EasyEDA editor*. Available at: <<https://easyeda.com/editor#id=b2d33aa2224e48c4ab4bcd51351e7376>>. Accessed in: 07/01/2023.
- [15] ESPRESSIF. *ESP32-DevKitS(-R)*. Available at: <<https://docs.espressif.com/projects/esp-idf/en/latest/esp32/hw-reference/esp32/user-guide-devkits-r-v1.1.html>>. Accessed in: 08/01/2023.
- [16] ESPRESSIF. *ESP8266*. Available at: <<https://www.espressif.com/en/products/socs/esp8266>>. Accessed in: 08/01/2023.
- [17] JLCPCB. *jlcpcb order*. Available at: <<https://jlcpcb.com/order>>. Accessed in: 08/01/2023.
- [18] FELIPEFOLP. *Sensor de fluxo*. Available at: <<https://www.filipeflop.com/produto/sensor-de-fluxo-de-agua-12-yf-s201/#tab-description>>. Accessed in: 08/01/2023.
- [19] SENSOR, A. *Bóia elétrica*. Available at: <<https://anauger.com.br/anauger-sensor-boia-de-nivel/>>. Accessed in: 29/01/2023.
- [20] RECICOMP. *Esp32 dev kit*. Available at: <<https://www.recicomp.com.br/produtos/modulo-wifi-esp32-bluetooth-30-pinos/>>. Accessed in: 29/01/2023.
- [21] RECICOMP. *Sensor de fluxo*. Available at: <<https://www.recicomp.com.br/produtos/sensor-de-fluxo-vazao-de-agua-1-2/>>. Accessed in: 29/01/2023.
- [22] LIVRE, M. *Bóia elétrica*. Available at: <<https://produto.mercadolivre.com.br/MLB-854158405-chave-boia-automatica-de-nivel-anauger-para-bomba-de-agua->

- JM#position=3&search_layout=stack&type=item&tracking_id=dcda90f9-5843-4748-8b00-8534c444d821>. Accessed in: 29/01/2023.
- [23] RECICOMP. *Relé 5V*. Available at: <<https://www.recicomp.com.br/produtos/rele-5vdc-10a-5-terminais/>>. Accessed in: 29/01/2023.
- [24] RECICOMP. *Diodo 1N4007 SMD*. Available at: <<https://www.recicomp.com.br/produtos/1n4007-smd-m7-diodo-retificador/>>. Accessed in: 29/01/2023.
- [25] RECICOMP. *Transistor BC547 SMD*. Available at: <<https://www.recicomp.com.br/produtos/mmbt2222-smd-transistor-npn/>>. Accessed in: 29/01/2023.
- [26] RECICOMP. *Conector KRE 3 vias*. Available at: <<https://www.recicomp.com.br/produtos/conector-borne-kre-3-vias/>>. Accessed in: 29/01/2023.
- [27] RECICOMP. *Conector KRE 2 vias*. Available at: <<https://www.recicomp.com.br/produtos/conector-borne-kre-2-vias/>>. Accessed in: 29/01/2023.
- [28] RECICOMP. *Resistor SMD 0805 1/8W 5%*. Available at: <<https://www.recicomp.com.br/produtos/resistor-smd-0805-1-8w-5-10-unidades/>>. Accessed in: 29/01/2023.
- [29] RECICOMP. *LED Difuso 3mm*. Available at: <<https://www.recicomp.com.br/produtos/led-difuso-3mm/>>. Accessed in: 29/01/2023.
- [30] PEDRO, D. *Trabalho de conclusão de curso (UFPE)*. Available at: <<https://www.youtube.com/watch?v=JBMJXgaoTfc>>.
- [31] ALIEXPRESS. *Ac 220v dc 12v 24v 1/2 "3/4" de plástico elétrico normalmente fechado válvula solenóide interruptor controlador de pressão de ar de água magnético*.