



**GOVERNO DO ESTADO DO RIO DE JANEIRO
SECRETARIA DE ESTADO DE CIÊNCIA, TECNOLOGIA E INOVAÇÃO - SECTI
FUNDAÇÃO DE APOIO À ESCOLA TÉCNICA - FAETEC
ESCOLA TÉCNICA ESTADUAL FERREIRA VIANA - ETEFV
COORDENAÇÃO DE ELETRÔNICA**

Christian F. Libório
Gabriel Lucas Martins
Kayke da S. Souza
Matheus N. Tesch
Thainá R. Pessanha

AquaScum

Rio de Janeiro
2025

Christian F. Libório
Gabriel Lucas Martins
Kayke da S. Souza
Matheus N. Tesch
Thainá R. Pessanha

AquaScum

Trabalho de Conclusão de Curso apresentado à Coordenação de Eletrônica da ETE Ferreira Viana, da Fundação de apoio à Escola Técnica, como requisito parcial para a obtenção da habilitação profissional de Técnico de Nível Médio em Eletrônica sob a orientação do Professor Luis Henrique Monteiro de Castro e do Professor Marcelo de Almeida Duarte.

Rio de Janeiro
2025

Prof. Luis Henrique M. de Castro
ETE Ferreira Viana
Orientador

Prof. Marcelo de Almeida Duarte
ETE Ferreira Viana
Orientador

Christian F. Libório
Gabriel Lucas Martins
Kayke da S. Souza
Matheus N. Tesch
Thainá R. Pessanha

AquaScum

Aprovada em : ____ / ____ / ____ **Conceito:** _____

Prof. XXXXXXXXXXXXXXXX
ETE Ferreira Viana
ID:

Prof. XXXXXXXXXXXXXXXX
ETE Ferreira Viana
ID:

Prof. XXXXXXXXXXXXXXXX
ETE Ferreira Viana
ID:

Rio de Janeiro
2025

RESUMO

LIBÓRIO, Christian; MARTINS, Gabriel; SOUZA, Kayke; TESCH, Matheus; PESSANHA, Thainá. *AquaScum*. 2025. Trabalho de Conclusão de Curso (em desenvolvimento) - Curso Técnico em Eletrônica, Escola Técnica Estadual Ferreira Viana, Fundação de Apoio à Escola Técnica, Rio de Janeiro, 2025.

O **Projeto AquaScum** será um sistema eletrônico pensado para monitorar e regular a temperatura e o pH da água em aquários de água doce, garantindo condições ideais para a saúde dos organismos aquáticos. O sistema utiliza sensores para coletar dados em tempo real e, com base neles, um microcontrolador realizará ajustes automáticos sempre que as condições saírem da faixa ideal. A temperatura será monitorada por um sensor de temperatura (DS18B20), enquanto alertas serão gerados sempre que o pH estiver fora do padrão, ativando as correções. Além disso, uma interface de controle, acessível via display LCD, permitirá o monitoramento. O projeto visa automatizar a manutenção do ambiente aquático, com planos para futuras melhorias, como a integração de funcionalidades que atendam a outras necessidades comuns na manutenção de aquários.

Palavras-chave: Arduino; Aquarismo; Automatização;

LISTA DE ABREVIATURAS/SIGLAS

LCD - *liquid crystal display*

pH - *escala que mede a acidez e a basicidade de uma solução*

LED - *light-emitting diode ou diodo emissor de luz*

AC - *corrente alternada*

DC - *corrente contínua*

Sumário

1. INTRODUÇÃO	6
1.1 OBJETIVO.....	7
1.2 ESCOPO DO TRABALHO.....	7
1.3 JUSTIFICATIVA DO TEMA.....	8
1.4 METODOLOGIA ADOTADA.....	8
2. MATERIAIS.....	9
3. DIAGRAMA DE FUNCIONAMENTO.....	10
4. EMBASAMENTO TEÓRICO	11
4.1 ARDUINO.....	11
4.2 SENSOR DE TEMPERATURA DS18B20.....	11
4.3 SENSOR DE PH	12
4.4 MINI VENTILADOR COOLER	12
4.5 SERPENTINA DE AQUECIMENTO.....	13
4.6 MÓDULO RELÉ DE 4 CANAIS	13
4.7 SENSOR DE NÍVEL LATERAL.....	14
4.8 MÓDULO WI-FI ESP8266.....	14
4.9 DISPLAY 20X4	15
4.10 MÓDULO RELÓGIO TEMPO REAL.....	15
5. DIFICULDADES ENFRENTADAS.....	16
6. CÓDIGO FONTE.....	16-20
7. MEDIÇÕES E RESULTADOS.....	21
8. CONCLUSÃO.....	21
9. REFERÊNCIAS.....	22

1.0 INTRODUÇÃO

A criação e manutenção de aquários demandam cuidados rigorosos com diversos parâmetros ambientais, sendo a temperatura e o pH da água dois dos fatores mais relevantes para garantir a saúde e o bem-estar dos organismos aquáticos. O controle inadequado dessas variáveis pode acarretar desequilíbrios no ecossistema do aquário, comprometendo a qualidade de vida dos peixes e demais seres vivos. Com o avanço da tecnologia e o crescente interesse por soluções automatizadas, tornou-se viável o desenvolvimento de sistemas capazes de monitorar e regular essas condições de forma contínua e precisa. Nesse cenário, a automação surge como uma aliada fundamental, permitindo a criação de ambientes estáveis com menor necessidade de intervenção manual. Além disso, sistemas automatizados contribuem para a redução de erros humanos, otimizam o tempo de manejo e promovem um ambiente mais seguro para os organismos aquáticos. O presente trabalho propõe o desenvolvimento do **Projeto AquaScum**, um sistema eletrônico automatizado voltado para o monitoramento e controle da temperatura e a indicação de variações de pH em aquários de água doce.

O pH de um aquário pode variar por diversos motivos, e essas mudanças afetam diretamente a saúde dos peixes e plantas. Por isso, o controle dele é essencial para os organismos ali presentes. Cada espécie tem uma faixa ideal de pH, e valores fora do recomendado podem causar estresse, doenças ou morte. Entre as principais causas para sua alteração estão o acúmulo de matéria orgânica, como restos de ração, fezes e urina, que acidificam a água com o tempo. Além disso, o pH influencia a toxicidade de substâncias como a amônia e a eficiência da filtragem biológica. Trocas de água também influenciam, pois a nova água pode ter um pH diferente. Elementos do aquário, como substratos, pedras, conchas ou troncos, liberam substâncias que podem aumentar ou diminuir o pH. Além disso, o uso de produtos químicos, a presença de dióxido de carbono (especialmente em aquários plantados) e a evaporação da água contribuem para alterações no pH. Sendo assim, para garantir um ambiente equilibrado é necessário seu monitoramento e ajustes. Já a temperatura do aquário é crucial porque afeta diretamente o metabolismo e o bem-estar dos peixes, o equilíbrio do ecossistema, a oxigenação da água e o desenvolvimento de bactérias benéficas. Mudanças bruscas ou temperaturas inadequadas podem causar estresse, facilitar doenças e até levar à morte dos peixes. Por isso, é fundamental manter uma temperatura estável e adequada às espécies mantidas no aquário, garantindo um ambiente saudável e equilibrado.

1.1 OBJETIVO

O projeto propõe o desenvolvimento de um sistema automatizado para monitoramento e controle da temperatura da água e notificação de variações críticas no pH, com o objetivo de promover um ambiente saudável e estável em aquários de água doce. Para isso, será projetado e implementado um circuito eletrônico utilizando sensores de temperatura e pH, conectados a um microcontrolador Arduino, responsável pelo monitoramento contínuo dos dados. O sistema acionará automaticamente sensor de temperatura sempre que necessário para manter a temperatura dentro dos parâmetros ideais. Além disso, contará com um sistema de alertas visuais e sonoros para notificar o usuário sobre qualquer alteração crítica nos parâmetros. Para facilitar a interação, será desenvolvida uma interface de controle acessível, utilizando um display LCD, permitindo o acompanhamento em tempo real das condições do aquário.

1.2 ESCOPO DO TRABALHO

Este projeto visa criar uma solução de baixo custo e fácil implementação para aquaristas amadores, focando na automação do controle da temperatura e do monitoramento do pH da água. O escopo deliberadamente exclui outros parâmetros, como oxigenação ou iluminação, para simplificar a implementação e torná-la acessível, sem a necessidade de equipamentos sofisticados.

Partindo da dificuldade comum em manter as condições estáveis, o projeto busca responder: como desenvolver um sistema automatizado, de baixo custo e fácil uso, para monitorar e ajustar variações de temperatura e pH em aquários de água doce?

1.3 JUSTIFICATIVA DO TEMA

A escolha deste tema justifica-se pela crescente popularização do aquarismo no Brasil, especialmente após o período de isolamento social ocasionado pela pandemia da Covid-19, que impulsionou o interesse da população por atividades domésticas e pelo cuidado com animais de estimação, como peixes ornamentais. Segundo dados do IBGE, cerca de 11 milhões de brasileiros estão envolvidos com o aquarismo (Squassoni, G.H., 2021). Entretanto, muitos aquaristas enfrentam dificuldades em manter os parâmetros ideais da água de forma contínua, o que pode comprometer a saúde dos peixes. Assim, o desenvolvimento de uma solução automatizada, acessível e eficiente contribui não apenas para o bem-estar dos animais, mas também para a disseminação de boas práticas entre os criadores. Além disso, o projeto possui relevância acadêmica por promover o estudo e aplicação de conceitos de automação, eletrônica embarcada e desenvolvimento de sistemas inteligentes.

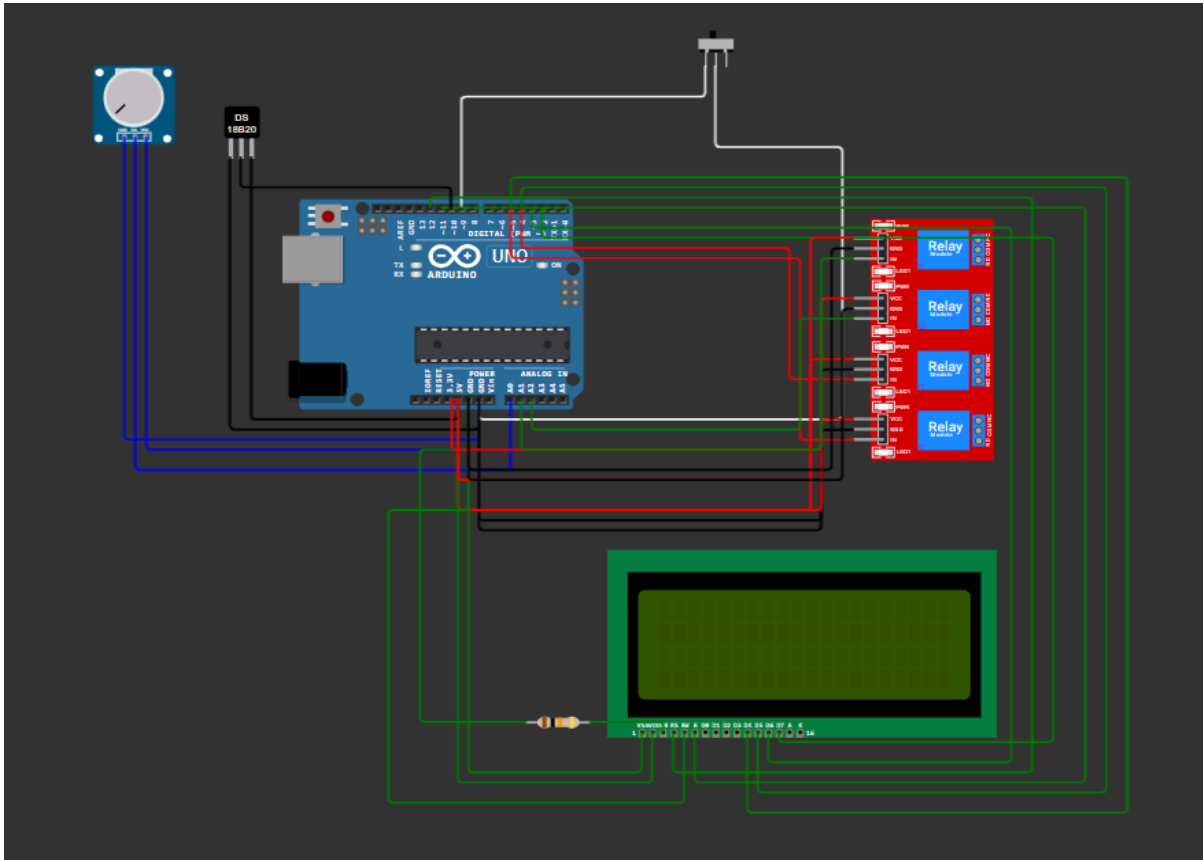
1.4 METODOLOGIA ADOTADA

A metodologia adotada caracteriza-se como uma pesquisa aplicada, de natureza experimental, com abordagem prática baseada na prototipagem de hardware. O desenvolvimento do sistema envolverá a utilização de sensores específicos para a medição de temperatura (como o DS18B20) e de pH, um microcontrolador (Arduino Uno) e diversos componentes auxiliares, como módulos de relé, resistores, display LCD e módulos de comunicação. Serão implementadas rotinas para a leitura dos sensores, controle de temperatura com acionamento automático de resistência elétrica. A interface de interação com o usuário será desenvolvida através de um display LCD e botões de navegação para acesso às funções do sistema. Durante o processo de desenvolvimento, serão conduzidos testes em ambiente controlado, com o objetivo de validar o funcionamento dos módulos, a precisão das medições e a resposta dos atuadores, garantindo a confiabilidade e a eficácia do sistema proposto.

2.0 MATERIAIS

PRODUTOS	CUSTOS	LINKS
ARDUÍNO UNO R3	R\$ 54,90	bitmaker
MÓDULO WI-FI ESP8266	R\$ 16,90	bitmaker
MINI VENTILADOR COOLER	R\$ 79,90	Shopee
SENSOR DE TEMPERATURA DS18B20	R\$ 14,90	bitmaker
MÓDULO RELÉ 4 CANAIS	R\$ 21,90	bitmaker
2 SENSOR DE NÍVEL LATERAL	R\$ 17,90 (x2R\$35,80)	bitmaker
DISPLAY LCD 20x4	R\$ 34,90	bitmaker
MÓDULO RELÓGIO TEMPO REAL RTC DS1302	R\$ 8,00	bitmaker
MÓDULO DE PH/ELETRODO	R\$ 65,00	Shopee
TAMPÃO DE PH 7.0	R\$ 30,00	Shopee
CABOS JUMPER (40 UNIDADES)	R\$ 11,90	bitmaker
CHAVES LIGA/DESLIGA (4 UNIDADES)	R\$ 10,00	bitmaker
CHAVES LIGA/DESLIGA (4 UNIDADES)	R\$ 10,00	bitmaker
CAIXA DE MDF	R\$ 20,00	Kalunga
TOTAL	R\$ 414,10	

3.0 DIAGRAMA DE FUNCIONAMENTO



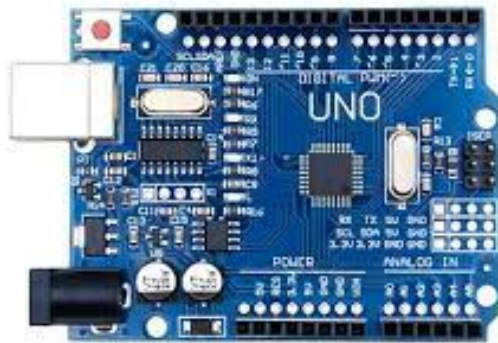
[LINK](#)

- **Switch** - representa a boia.
- **Potenciômetro** - representa o sensor de pH.

4.0 EMBASAMENTO TEÓRICO

4.1 ARDUINO

o **Arduino** é uma série de microcomputadores de placa única com componentes integrados. Série de plataformas programáveis de prototipagem eletrônica de placa única e hardware livre (código aberto), que permite aos usuários criar objetos eletrônicos interativos e independentes, usando o microcontrolador Atmel AVR ou ARM com suporte de entrada/saída embutido. A plataforma foi criada em 2005 na Itália, com o objetivo de criar ferramentas de baixo custo, acessíveis, flexíveis, independentes e de fácil uso para principiantes, amadores e profissionais, com foco especial naqueles que não têm acesso a controladores sofisticados e ferramentas complexas.



4.2 SENSOR DE TEMPERATURA DS18B20

O **DS18B20** é um tipo de sensor que fornece leituras de temperatura de 9 a 12 bits. Esses valores mostram a temperatura de um determinado dispositivo ou ambiente. A comunicação deste sensor pode ser feita através de um protocolo de barramento de um fio que usa uma linha de dados para se comunicar com um microprocessador interno. Além disso, este sensor obtém a fonte de alimentação diretamente da linha de dados, de modo que a necessidade de uma fonte de alimentação externa pode ser eliminada.



4.3 SENSOR DE PH

O **Sensor de pH Arduino** é um dispositivo de medição utilizado em diversos setores, capaz de fazer a medição da acidez e da basicidade de um determinado líquido, mostrando através de uma escala fixa se o líquido está em condições de um elemento base, ou ácido, sendo 7 considerado um pH de elemento neutro, nem ácido, nem básico.



4.4 MINI VENTILADOR COOLER

Um **mini ventilador cooler** é um dispositivo projetado para resfriar e dissipar calor em sistemas eletrônicos, como computadores, e equipamentos elétricos, além de ser usado em projetos de eletrônica e em equipamentos de refrigeração como choques de aquecimento. Também existem versões portáteis para uso pessoal ou em ambientes fechados, alguns até com umidade.



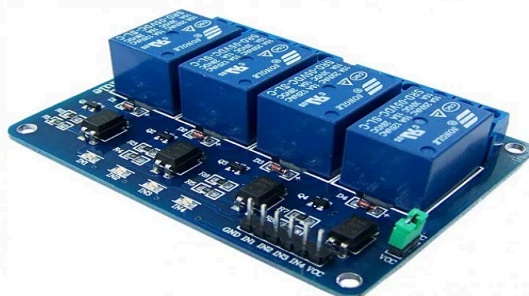
4.5 SERPENTINA DE AQUECIMENTO

A **serpentina de aquecimento** composta por uma resistência interna a serpentina será alimentada quando o sensor de temperatura detectar uma temperatura abaixo de 25° fazendo com que a resistência seja alimentada por poucos segundos até que a água do aquário esteja em uma temperatura razoável.



4.6 MÓDULO RELÉ DE 4 CANAIS

O **módulo relé de 4 canais** permite que um microcontrolador, como Arduino ou ESP32, controle até quatro dispositivos elétricos de maior potência (como lâmpadas, bombas e ventiladores) usando sinais de baixa tensão. Cada relé funciona como um interruptor eletrônico com três terminais: COM, NO (normalmente aberto) e NC (normalmente fechado). Quando o microcontrolador envia um sinal, o relé é ativado e liga ou desliga o circuito da carga conectada. O módulo geralmente opera com 5V (ou 3,3V em algumas versões), suporta até 10A em 250V AC ou 30V DC e pode ter optoacopladores para maior segurança elétrica. É amplamente usado em automação residencial e industrial, como controle de iluminação, bombas d'água e sistemas de aquário. É importante garantir alimentação adequada, isolamento entre os circuitos e compatibilidade de tensão com o microcontrolador.



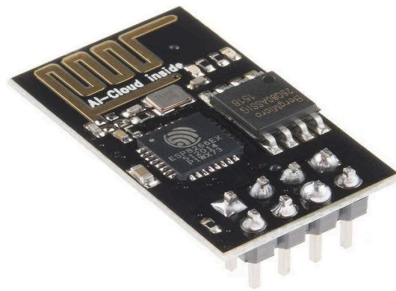
4.7 SENSOR DE NÍVEL LATERAL

O **sensor de nível lateral** é um dispositivo usado para detectar o nível de líquidos em tanques ou reservatórios, sendo instalado na lateral do recipiente, no ponto exato que se deseja monitorar (nível máximo ou mínimo). Ele funciona por meio de um flutuador magnético que se move conforme o nível do líquido sobe ou desce. Quando o flutuador atinge determinada posição, ele abre ou fecha um contato elétrico, enviando um sinal que pode ser usado por controladores como Arduino ou ESP32 para acionar bombas, válvulas ou alarmes.



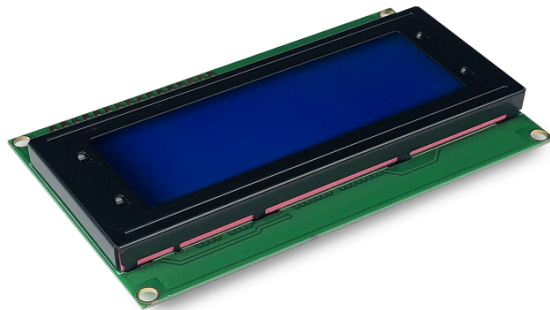
4.8 MÓDULO WI-FI ESP8266

O **módulo ESP8266** é uma solução compacta, econômica e poderosa para adicionar conectividade WiFi aos seus projetos, sendo ideal para o seu sistema de automação de aquário com controle remoto/internet, caso você queira que seu controlador (como o ESP32 ou outro) envie dados pela rede ou receba comandos via WiFi.



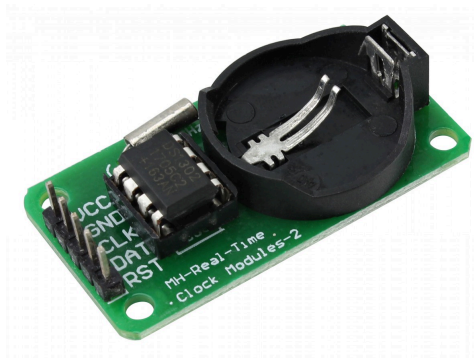
4.9 DISPLAY LCD 20X4

O **display LCD 20x4** é um módulo utilizado para exibir informações em tempo real, como temperatura, pH, nível de água e status de sistemas em projetos com microcontroladores, como Arduino e ESP32. Ele possui 20 colunas e 4 linhas, permitindo mostrar até 80 caracteres simultaneamente. Geralmente utiliza o controlador HD44780 ou compatíveis e pode se comunicar com o microcontrolador por interface paralela (4 ou 8 bits) ou por I²C, que reduz o número de fios de conexão. O módulo é alimentado normalmente com 5V, possui retroiluminação (backlight) em LED e ajuste de contraste por potenciômetro. O display é ideal para painéis informativos em sistemas automatizados, oferecendo boa legibilidade e baixo consumo. É importante garantir alimentação estável, ajustar corretamente o contraste, e verificar a compatibilidade de tensão com o microcontrolador, já que o ESP 32 opera em 3,3V.



4.10 MÓDULO RELÓGIO TEMPO REAL RTC DS1302

O **módulo DS1302** é um circuito integrado que funciona como um relógio em tempo real (RTC) de baixo consumo, amplamente utilizado em projetos com Arduino e microcontroladores. Suas principais características incluem operação com baixa energia, capacidade de backup por bateria (como CR2032) para manter o tempo sem energia principal, comunicação serial via 3 fios (CE, I/O e SCLK), cristal interno de 32.768 kHz para precisão temporal, e 31 bytes de RAM adicional para armazenamento de dados do usuário. Ele armazena informações de data e hora em formato BCD, sendo uma solução econômica e de fácil integração para aplicações como relógios digitais, sistemas de datalogging e automação residencial.



5. DIFICULDADES ENCONTRADAS

Enfrentamos uma série de contratempos significativos durante a execução do projeto, com destaque para os desafios de montagem e integração dos componentes. O mau contato no protoboard foi um dos principais obstáculos, pois gerava falhas que inicialmente eram atribuídas ao código. Esse equívoco nos levava a ajustes desnecessários no código, até que fosse identificado o problema de montagem. O processo então se repetia: o líder do projeto precisava desmontar e remontar todo o conjunto para assegurar seu funcionamento, mas um novo fio solto ou mau contato frequentemente dava início a um ciclo vicioso de retrabalho e desorganização, consumindo um tempo valioso.

No que diz respeito aos sensores, a maior complexidade esteve na implementação do sensor de pH. Sua calibração apresentou dificuldades persistentes, resultando em leituras inconsistentes e fora dos parâmetros esperados. Curiosamente, o código de calibração funcionava em ambientes de teste, mas falhava quando integrado ao nosso sistema principal. Em contrapartida, o sensor de temperatura apresentou poucos problemas intrínsecos; suas falhas estavam principalmente associadas aos já citados problemas de mau contato no protoboard e à integração com o outro sensor.

Outro imprevisto crítico foi a aquisição de um modelo de ventilador inadequado para a aplicação. Como consequência, seu desempenho de arrefecimento ficou abaixo do esperado e sua integração ao circuito mostrou-se problemática.

Diante dessas sucessivas dificuldades e para viabilizar a conclusão do projeto dentro do prazo, foi necessário tomar a decisão de simplificar o escopo original. Dessa forma, funcionalidades como o monitoramento do nível de água e o controle de certos parâmetros tiveram que ser abandonadas.

6. CÓDIGO FONTE

```
#include <LiquidCrystal.h>
#include <OneWire.h>
#include <DallasTemperature.h>

// Definições de pinos
#define ONE_WIRE_BUS 10
#define RELAY_HEATER 7 // Aquecedor
#define RELAY_FAN 6 // Ventilador
#define RELAY_PH_UP A1 // Dosador alcalino
#define RELAY_PH_DOWN A2 // Dosador ácido
#define FLOAT_SWITCH_PIN 13
#define BUTTON_PIN 8

// Inicializa barramento 1-Wire
```

```

OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);

// Inicializa o LCD: RS, E, D4, D5, D6, D7
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

// Variáveis de controle
float temperature = 0.0;
int tempMin = 26;
int tempMax = 28;
float phMin = 6.5;
float phMax = 7.5;

float calibracao_ph7 = 2.5; // Tensão obtida em solução de calibração pH 7
float calibracao_ph4 = 3.06; // Tensão obtida em solução de calibração pH 4
float calibracao_ph10 = 0.00; // Tensão obtida em solução de calibração pH 10

#define UTILIZAR_PH_10 false // Habilita calibração entre pH 7 e 10, caso contrário
utiliza pH 7 e 4.
float m;
float b;
int buf[10];

bool buttonState = false;
bool lastButtonState = false;

unsigned long lastScreenSwitch = 0;
const unsigned long screenInterval = 5000;
bool showPhLimits = false;
int nivel;

void setup() {
  Serial.begin(9600);
  sensors.begin();
  lcd.begin(20, 4);

  pinMode(RELAY_HEATER, OUTPUT);
  pinMode(RELAY_FAN, OUTPUT);
  pinMode(RELAY_PH_UP, OUTPUT);
  pinMode(RELAY_PH_DOWN, OUTPUT);
  pinMode(BUTTON_PIN, INPUT_PULLUP);
  pinMode(FLOAT_SWITCH_PIN, INPUT_PULLUP);

```

```

digitalWrite(RELAY_HEATER, LOW);
digitalWrite(RELAY_FAN, LOW);
digitalWrite(RELAY_PH_UP, LOW);
digitalWrite(RELAY_PH_DOWN, LOW);

if (UTILIZAR_PH_10) {
  m = (7.0 - 10.0) / (calibracao_ph7 - calibracao_ph10);
  b = 10.0 - m * calibracao_ph10;
} else {
  m = (4.0 - 7.0) / (calibracao_ph4 - calibracao_ph7);
  b = 7.0 - m * calibracao_ph7;
}
}

void loop() {
  // Leitura do botão para alternar limites de temperatura
  buttonState = digitalRead(BUTTON_PIN);
  if (buttonState == LOW && lastButtonState == HIGH) {
    if (tempMin == 26) {
      tempMin = 18;
      tempMax = 22;
    } else {
      tempMin = 26;
      tempMax = 28;
    }
  }
}
lastButtonState = buttonState;

// Leitura da temperatura
sensors.requestTemperatures();
float tempRead = sensors.getTempCByIndex(0);
if (tempRead != -127.00) {
  temperature = tempRead;
} else {
  Serial.println("Erro na leitura da temperatura");
}

// Leitura do pH simulado via potenciômetro em A0
for (int i = 0; i < 10; i++) {
  buf[i] = analogRead(A0);
  delay(10);
}

```

```
}
```

```
// Ordena em ordem crescente
```

```
for (int i = 0; i < 9; i++) {  
  for (int j = i + 1; j < 10; j++) {  
    if (buf[i] > buf[j]) {  
      int temp = buf[i];  
      buf[i] = buf[j];  
      buf[j] = temp;  
    }  
  }  
}
```

```
int valorMedio = 0;  
for (int i = 2; i < 8; i++) {  
  valorMedio += buf[i];  
}
```

```
float tensao = (valorMedio * 5.0) / 1024.0 / 6;  
float ph = m * tensao + b;  
Serial.println(ph);
```

```
// Controle do LCD  
lcd.clear();  
lcd.setCursor(0, 0);  
lcd.print("Temp:");  
lcd.print(temperature);  
lcd.print("C");
```

```
lcd.setCursor(12, 0);  
lcd.print("pH:");  
lcd.print(ph);
```

```
lcd.setCursor(0, 1);  
lcd.print("TempMin:");  
lcd.print(tempMin);  
lcd.print("C");
```

```
lcd.setCursor(12, 1);  
lcd.print("Max:");  
lcd.print(tempMax);  
lcd.print("C");
```

```
lcd.setCursor(0, 2);  
lcd.print("pHMin:");  
lcd.print(phMin);
```

```
lcd.setCursor(11, 2);  
lcd.print("Max:");  
lcd.print(phMax);
```

```
// Controle de temperatura  
if (temperature < tempMin) {  
    digitalWrite(RELAY_HEATER, HIGH);  
    digitalWrite(RELAY_FAN, LOW);  
    Serial.println("Aquecedor LIGADO");  
} else if (temperature > tempMax) {  
    digitalWrite(RELAY_HEATER, LOW);  
    digitalWrite(RELAY_FAN, HIGH);  
    Serial.println("Ventilador LIGADO");  
} else {  
    digitalWrite(RELAY_HEATER, LOW);  
    digitalWrite(RELAY_FAN, LOW);  
    Serial.println("Temperatura OK");  
}
```

```
// Controle de pH  
if (ph < phMin) {  
    digitalWrite(RELAY_PH_UP, HIGH);  
    digitalWrite(RELAY_PH_DOWN, LOW);  
    Serial.println("Corrigindo pH BAIXO");  
} else if (ph > phMax) {  
    digitalWrite(RELAY_PH_UP, LOW);  
    digitalWrite(RELAY_PH_DOWN, HIGH);  
    Serial.println("Corrigindo pH ALTO");  
} else {  
    digitalWrite(RELAY_PH_UP, LOW);  
    digitalWrite(RELAY_PH_DOWN, LOW);  
    Serial.println("pH OK");  
}
```

```
delay(1000)
```

7. Medições e Resultados

Durante os testes realizados no sistema, foram definidos os parâmetros ideais para garantir o bem-estar dos peixes Beta no aquário. A temperatura da água deve permanecer entre **26 °C e 28 °C**, faixa considerada ideal para a maioria das espécies tropicais. Quando a temperatura cai abaixo de 26 °C, o relé aciona o aquecedor automaticamente, elevando a temperatura. Caso ultrapasse 28 °C, o relé aciona o ventilador e o próprio é ativado para resfriar a água, mantendo o equilíbrio térmico.

Em relação ao pH, os valores foram ajustados para uma faixa de **6,5 a 7,5**, próxima ao pH neutro, adequado para a manutenção da vida aquática. Se o pH ficar abaixo de 6,8, o sistema aciona o dosador de solução alcalina (pH Up) para corrigir a acidez. Quando o pH ultrapassa 7,5, é acionado o dosador de solução ácida (pH Down), restabelecendo o equilíbrio.

Nas medições práticas, foi possível calibrar o eletrodo de pH com soluções-padrão, e o sensor apresentou leituras estáveis em torno de pH 7,0, confirmando a precisão do sistema de monitoramento e controle automatizado.

8. CONCLUSÃO

O desenvolvimento deste projeto de aquário automatizado com Arduino proporcionou uma experiência prática significativa na aplicação de conceitos de automação, eletrônica e programação embarcada. Foi possível implementar e testar com êxito algumas das principais funcionalidades do sistema, como o monitoramento de temperatura e de pH.

Por se tratar de um projeto complexo, nem todas as funções previstas puderam ser implementadas em tempo hábil. Porém, temos planos de longo prazo para o desenvolvimento do projeto, com o objetivo de aprimorá-lo em versões futuras.

Em suma, o projeto reforça o potencial da automação aplicada a pequenos ecossistemas, mostrando que soluções acessíveis podem ser criadas com tecnologias de baixo custo e boa eficiência.

9. REFERÊNCIAS

COELHO, Alessandra Dutra et al. Aquário Automatizado. São Caetano do Sul: Instituto Mauá de Tecnologia, 2019. Disponível em: <https://maua.br/files/artigos/aquario-automatizado.pdf>. Acesso em: [colocar data]. [Instituto Mauá de Tecnologia](#)

SOUZA, Pedro Henrique Arantes de et al. Automação e Monitoramento em Aquários utilizando Arduino. Revista Interação, v. 19, n. 2, p. 162-181, 2017. Disponível em: <https://periodicos.unis.edu.br/interacao/article/download/144/130/534>. Acesso em: [colocar data]. [Periódicos UNIS](#)

SANTOS, Rodrigo Jeremias dos et al. Estudo de Sistema de Controle Automático para Aquário utilizando uma Plataforma de Prototipagem Eletrônica Open Source. Revista Ibero-Americana de Humanidades, Ciências e Educação, v. 8, n. 4, p. 569-586, 2022. DOI:10.51891/rease.v8i4.4852. [Periódico Rease](#)

PINZETTA, Giovani Begnini; FIGUEIREDO, José Antônio Oliveira de. Computação Embarcada na Automatização de Aquários Marinhos. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) – Instituto Federal Sul-rio-grandense, Campus Passo Fundo, 2022. Disponível em: <https://painel.passofundo.ifsul.edu.br/uploads/arq/202302071428571963938365.pdf>. Acesso em: [colocar data]. [Painel Passo Fund.](#)