

Criação de Camarão em Cativeiro com Monitoramento IOT para Gastronomia

Captive Shrimp Farming with IOT Monitoring for Gastronomy

Kusaka.J	{ joao.kusaka@fatec.sp.gov.br }
Muniz.L	{ leandro.muniz@fatec.sp.gov.br }
Ferrari.M	{ matheus.abrahao@fatec.sp.gov.br }
Rodrigues.T	{ tiago.rodrigues39@fatec.sp.gov.br }
Roder.V	{ victor.roder@fatec.sp.gov.br }

RESUMO

A carcinicultura é uma atividade de grande relevância econômica e ambiental no Brasil, com destaque para o Nordeste, onde o país figura entre os principais produtores de camarão. Contudo, a criação enfrenta desafios consideráveis, como a má gestão dos tanques, o que pode resultar em poluição, aumento de doenças e, consequentemente, uma queda na qualidade do produto. Para mitigar esses problemas e ao mesmo tempo promover práticas sustentáveis alinhadas aos Objetivos de Desenvolvimento Sustentável (ODS) 12 e 14, propõe-se um sistema de monitoramento contínuo, que combina arduínos e Inteligência Artificial. Esse sistema monitora parâmetros essenciais da água, incluindo temperatura, pH e níveis de amônia, além de automatizar a alimentação dos camarões de forma precisa e eficiente. Os dados coletados são armazenados, permitindo análises históricas e melhorias contínuas no manejo dos cativeiros. A plataforma web oferece aos usuários acesso em tempo real aos dados de cada tanque, além de funcionalidades para gerenciamento de informações e controle automatizado da alimentação, garantindo maior comodidade e precisão na operação. Com esses recursos, o sistema propõe uma solução que visa não apenas melhorar a eficiência da produção, mas também reduzir o impacto ambiental, contribuindo para uma carcinicultura mais responsável e sustentável, e apresentando-se como alternativa de grande potencial para produtores em outras regiões do país e no exterior.

PALAVRAS-CHAVE: Carcinicultura; Monitoramento; Sustentável.

ABSTRACT

Shrimp farming is an economically and environmentally significant activity in Brazil, particularly in the Northeast, where the country stands out as one of the major shrimp producers. However, this industry faces considerable challenges, such as poor tank management, which can lead to water pollution, disease proliferation, and a decline in shrimp quality. To address these issues and promote sustainable practices aligned with Sustainable Development Goals (SDGs) 12 and 14, a continuous 24-hour monitoring system is proposed, combining Arduinos and Artificial Intelligence (AI). This system monitors essential water parameters, including temperature, pH, and ammonia levels, while automating shrimp feeding with precision and efficiency. The collected data is stored, enabling historical analyses and ongoing improvements in tank management. The web platform associated with the system provides users with real-time access to data from each tank, as well as management features and automated feeding control, ensuring greater convenience and operational accuracy. Through these features, the system offers a high-quality solution designed not only to improve production efficiency but also to reduce environmental impact, contributing to a more responsible and sustainable shrimp farming industry. This innovative approach presents significant potential for adoption by producers in other regions of the country and abroad.

KEYWORDS: Shrimp Farming; Monitoring; Sustainability.

INTRODUÇÃO

Em 2015, os 193 líderes mundiais comprometeram-se com os Objetivos de Desenvolvimento Sustentável (ODS) (The Global Goals, s.d.), estabelecidos na Agenda 2030, um plano de ação global voltado à erradicação da pobreza, à proteção do meio ambiente e à promoção da paz e prosperidade em todas as regiões do planeta.

Entre esses objetivos, destacam-se o ODS 12 (Produção e Consumo Sustentáveis) e o ODS 14 (Vida na Água). O ODS 12 busca garantir padrões sustentáveis de produção e consumo, assegurando

a saúde humana e reduzindo os impactos ambientais. Já o ODS 14 tem como meta a conservação e o uso sustentável dos oceanos, mares e recursos marinhos, combatendo a poluição, a degradação de habitats e a disseminação de doenças.

Dentro desse contexto de sustentabilidade e preservação dos ecossistemas aquáticos, destaca-se a carcinicultura, atividade aquícola voltada à criação de camarões. Segundo Bertini (Bertini et al., 2021), Caridea é um grupo de crustáceos amplamente conhecido como camarões, pertencentes à ordem Decapoda, e encontrados tanto em ambientes de água doce quanto salgada. Estima-se a existência de aproximadamente 3.500 espécies, distribuídas em 16 superfamílias e 36 famílias, das quais cerca de 800 vivem em água doce e 2.700 em água salgada. A carcinicultura, por sua vez, refere-se à técnica de criação desses animais em viveiros, podendo ser aplicada a espécies marinhas ou dulcícolas.

De acordo com Rocha (Rocha, 2023), a produção mundial de camarão tem crescido a uma taxa média de 4,97

No Brasil, o camarão representa a segunda maior fonte de valor na produção aquícola. O estado do Ceará lidera a produção nacional, com aproximadamente 43.778,493 toneladas registradas em 2022. A região Nordeste responde por cerca de 99

O Vale do Ribeira, localizado ao sul do estado de São Paulo, destaca-se como uma região propícia à carcinicultura em razão de suas condições ambientais favoráveis — rios, estuários e manguezais — que contribuem para o desenvolvimento sustentável da atividade. O setor é impulsionado pelo turismo e pela comercialização direta com restaurantes, fazendas e consumidores locais, sendo Cananéia, Eldorado e Sete Barras as principais cidades produtoras.

Os carídeos dulcícolas presentes no Vale do Ribeira pertencem principalmente às famílias Palaemonidae e Atyidae. A família Palaemonidae compreende cerca de 980 espécies, distribuídas entre as subfamílias Pontoniinae e Palaemoninae, com destaque para os gêneros Palaemon, Palaemonetes e Macrobrachium. Já a família Atyidae é composta por aproximadamente 469 espécies, distribuídas em 42 gêneros (Bertini et al., 2021).

Apesar do grande potencial produtivo, a carcinicultura ainda enfrenta desafios significativos, como a má gestão dos tanques e deficiências de infraestrutura, que podem resultar em poluição da água, acúmulo de resíduos e proliferação de doenças. Esses fatores comprometem a qualidade do produto final e impactam negativamente produtores e consumidores.

Para superar tais desafios e promover práticas mais eficientes e sustentáveis, torna-se essencial o investimento em tecnologias inteligentes voltadas ao monitoramento e controle ambiental. Nesse contexto, uma abordagem promissora é a implementação de um sistema de monitoramento inteligente baseado em Internet das Coisas (IoT).

O sistema proposto visa monitorar de forma eficiente as condições dos tanques de criação de camarões, acompanhando parâmetros críticos como temperatura, pH, níveis de oxigênio dissolvido e amônia. Sensores especializados conectados a microcontroladores, como o Arduino com conectividade Wi-Fi, realizam a coleta e transmissão dos dados em tempo real, garantindo o controle preciso do ambiente aquático.

A solução inclui também um aplicativo móvel, que oferece ao usuário uma interface gráfica intuitiva para visualização dos dados e controle remoto das operações. Além disso, será integrado um alimentador automático programável, permitindo definir horários e quantidades específicas de ração, com sensores de nível que otimizam o fornecimento de alimento.

Como diferencial, o sistema incorpora um mecanismo de recomendação inteligente, capaz de fornecer orientações personalizadas ao carcinicultor com base nos dados coletados. Esse recurso possibilita a geração de relatórios históricos em PDF, gráficos analíticos dos sensores, previsão de

variações nos parâmetros e notificações preventivas em situações críticas, como alterações bruscas de temperatura ou níveis inadequados de alimentação. Dessa forma, o sistema atua não apenas no monitoramento, mas também na antecipação de riscos e apoio à tomada de decisão, promovendo maior eficiência e segurança na produção.

Portanto, este trabalho propõe o desenvolvimento de um sistema inteligente de monitoramento e recomendação para a carcinicultura, alinhado aos Objetivos de Desenvolvimento Sustentável, especialmente os ODS 12 e 14, com o intuito de otimizar o manejo produtivo, reduzir impactos ambientais e impulsionar a sustentabilidade da atividade no Vale do Ribeira.

OBJETIVO

Este sistema de monitoramento tem como meta desenvolver um software que irá auxiliar os produtores na criação de camarões de alta qualidade e na manutenção eficiente do controle dos cativeiros. Dentre os objetivos específicos, destacam-se:

1. Monitorar remotamente os cativeiros, garantindo um ambiente aquático de alta qualidade para o desenvolvimento dos camarões.
2. Controlar continuamente os níveis de pH, temperatura e amônia da água, emitindo alertas automáticos sempre que os parâmetros ultrapassarem os limites ideais.
3. Registrar e armazenar dados históricos dos cativeiros para possibilitar análises comparativas e avaliação da qualidade ao longo do tempo.
4. Automatizar o fornecimento de ração, controlando horários e quantidades conforme parâmetros definidos pelo usuário.
5. Fornecer relatórios e gráficos gerados a partir das informações monitoradas, permitindo uma análise detalhada e fundamentada das condições do cultivo.
6. Cadastrar usuários, cativeiros e pontos de monitoramento, facilitando a gestão e o acesso centralizado às informações.
7. Desenvolver uma aplicação móvel intuitiva e acessível, proporcionando uma experiência prática e eficiente ao carcinicultor.
8. Contribuir para a sustentabilidade e eficiência da produção aquícola, reduzindo desperdícios e impactos ambientais.

ESTADO DA ARTE

A realização de um sistema de monitoramento para cuidado dos camarões é de grande importância, ajuda tanto o produtor a vender um produto de qualidade, tanto o consumidor que irá consumir algo de qualidade sem prejudicar a sua saúde. Por conta desta importância, é de se imaginar que já possam existir sistemas parecidos. Dentre os projetos já existentes, há de se destacar determinados artigos.

O trabalho(Dantas, 2020) é proposto um sistema de telemonitoramento e automação baseado em rede LoRaWAN. O artigo descreve o desenvolvimento de um sistema de monitoramento de temperatura e pH e automação de viveiros de camarão utilizando tecnologia LoRa. A plataforma visa auxiliar carcinicultores, engenheiros de pesca e colaboradores das fazendas de criação no cultivo de camarões. A comunicação entre os nós sensores e o gateway é realizada via protocolo LoRaWAN, permitindo a transferência de dados para a nuvem através de um broker MQTT.

Os resultados registraram as médias de cada setor trabalhado. A temperatura obteve a média de 0 a 80°C. O pH apresentou uma média entre 6,64 e 7,5.

(Uddin et al., 2020) diz a respeito de um sistema de monitoramento de fazendas de camarão

de água doce baseado em IoT. O sistema tem a proposta de monitorar a qualidade da água como temperatura, pH, salinidade, oxigênio dissolvido e turbidez e verificar o peso, tamanho e porcentagem de sobrevivência por meio de IoT. Isso porque envolve a coleta de dados de sensores em tempo real, armazenamento desses dados na nuvem e disponibilização de visualização e análise desses dados por meio de uma aplicação web. A camada física do sistema utiliza sensores para coletar dados e uma placa arduíno para processamento e transmissão desses dados para a nuvem.

Os resultados indicaram que o sistema funciona de maneira eficiente, com bons níveis de precisão, a temperatura obteve 98,%, nível de pH obteve 98,24%, salinidade obteve 94,12%, oxigênio obteve 95,46%, turbidez obteve 93,55%. Os camarões atingiram tamanhos entre 9 e 12 cm, peso varia entre 10 e 20g e a taxa de sobrevivência obteve 90%.

(Zainuddin; Idris; Azis, 2019) apresenta um sistema de monitoramento da qualidade da água para cultivo de camarão *Vannamiae* baseado em rede de sensores sem fios. O sistema proposto consiste em sensores para monitorar temperatura, pH e turbidez da água, integrados a microcontroladores e dispositivos de comunicação sem fio em conjunto de IoT. Os resultados do estudo incluem aspectos de hardware e software, destacando a produção de unidades transmissoras e receptores, bem como os testes de calibração dos sensores.

Os resultados indicam que o sistema pode funcionar bem, com altos níveis de precisão nos sensores de temperatura com 97,76%, pH com 98,85% e turbidez com 99,73%.

Em geral, os três projetos se assemelham em diversos fatores, como o monitoramento de temperatura e pH da água, a criação de um sistema de monitoramento e o uso de IoT. Um dos três artigos citados trabalha acerca do peso e tamanho do camarão.

Em comparação com os artigos anteriores, pode-se notar a diferença encontrada em nosso projeto. Ao invés de apenas o uso de IoT, contará com o uso de IA para auxiliar em tarefas como coleta de dados pelo hardware, criação de relatórios a partir das informações coletadas, analisar os dados para alertas sobre possíveis riscos caso ocorra uma mudança tanto nas condições dos tanques tanto na alimentação. Irá controlar a alimentação dos camarões para uma alimentação saudável com comestíveis apropriados para seu consumo, sua alimentação irá se basear em estimativas de crescimento e sobrevivência dos camarões nos cativeiros, caso não sejam controlados adequadamente, geram perdas econômicas e problemas nos cativeiros. Irá monitorar os níveis de amônia e irá realizar comparações com outros cativeiros.

METODOLOGIA

Neste projeto, foram utilizadas ferramentas e técnicas para auxiliar no desenvolvimento. Estas abordagens incluem o uso do Figma, uma plataforma colaborativa para design de interfaces e protótipos pertencente à empresa Figma, INC, do Modelo de Negócios Canvas, uma ferramenta de planejamento estratégico, do DER (Diagrama de Entidade e Relacionamento), os Diagramas de UML (Linguagem de Modelagem Unificada), o Diagrama de Rede.

O Figma é empregado como uma plataforma colaborativa de design para desenvolver os protótipos da interface do usuário do aplicativo. Isso possibilitará que a equipe de design trabalhe de forma conjunta, recebendo feedback instantâneo. Dessa forma, asseguraremos que a interface do usuário seja intuitiva e visualmente atrativa (Lopes, 2023).

O Modelo de Negócios Canvas é uma representação detalhada e abrangente da operação, receita e valor que uma empresa proporciona aos seus clientes. Ele visa auxiliar no desenvolvimento da percepção para entendimento da vida em sociedade o papel que lhes é atribuído. No caso específico, o canvas será utilizado para criar uma representação visual do projeto dentro do mercado gastronômico de camarões. Isso ajudará a equipe a identificar parceiros-chave, recursos-chave, atividades-chave

e fontes de receita. O objetivo é assegurar que o projeto esteja alinhado tanto com os objetivos científicos quanto com os de negócios(Biava, 2017).

O Diagrama de Entidade e Relacionamento (DER) é uma ferramenta fundamental na modelagem de dados, empregada para representar entidades, seus atributos e os relacionamentos entre elas em um sistema. As entidades são objetos do sistema que possuem atributos, que são características ou propriedades que as descrevem. Os relacionamentos, por sua vez, indicam as conexões ou associações entre essas entidades. No contexto do desenvolvimento do aplicativo, o DER será utilizado para estruturar o banco de dados, definindo entidades como doenças, análises, resultados obtidos, usuários, entre outras, e especificando os relacionamentos entre elas. Essa modelagem é crucial para garantir a eficiência no armazenamento e na recuperação de dados relacionados às doenças e às imagens associadas(Awari, 2023). As modelagens conceitual e lógico foi realizada por meio da plataforma BrModelo, enquanto o modelo físico foi realizado pela plataforma DbDiagram.io.

O LucidChart é utilizado para criar a representação de diagramas como: **Diagrama de classe**, **Diagrama de objetos** e **Diagrama de caso de uso**, além de fluxogramas, workflows e mapas mentais. A ferramenta auxilia para visualizar ideias, processos e sistemas mais complexos, permitindo maximizar os processos de transformação digital, gerenciando o conteúdo em nuvem e em servidores próprios.

OS Diagramas de UML são funcionais para apresentar os limites, a estrutura, o comportamento e os objetos dentro deles de forma geral. Os diagramas que serão usados são: **Diagrama de classe**, **Diagrama de objetos** e **Diagrama de caso de uso**(LucidChart, 2024b). O Diagrama de Classe mapeiam de uma forma clara e objetiva suas classes, atributos, operações e relações entre objetos(LucidChart, 2024c). O Diagrama de Objetos consiste em inserir atributos de um conjunto de objetos e apresentar a maneira em que se relacionam(LucidChart, 2024d). O Diagrama de Caso de Uso apresenta os detalhes dos usuários em seu sistema e as interações deles no sistema(LucidChart, 2024a).

HTML, CSS, JAVASCRIPT e BOOTSTRAP

Será utilizado o HTML (Linguagem de Marcação de Hipertexto) para estruturar e organizar o conteúdo da página web. Um hipertexto é um texto usado para fazer referência a outros textos. com o HTML, os usuários podem criar e estruturar seções, parágrafos e links usando elementos, tags e atributos(Longen, 2023). O CSS (Folha de Estilo em Cascata) é uma linguagem de estilo que foi usada para definir a aparência visual de uma página web. Ela é comumente utilizada para atribuir cores, fontes, tamanhos de texto e layouts. O CSS permite que nós separemos a aparência visual da página do conteúdo HTML, permitindo que criemos páginas web mais estéticas(G, 2022).

O Javascript é uma linguagem de programação de alto nível que será utilizada para criar interatividade na web. Ele usualmente é incrementado para criar efeitos animados, mapas, gráficos, menus drop-down entre outros. O projeto atual utiliza HTML, CSS e JavaScript para desenvolver o site da equipe e do projeto de forma responsiva e interativa, onde apresentamos a nossa equipe como, nossos objetivos, integrantes e os projetos que estão em desenvolvimento, a partir disso o usuário poderá conhecer melhor o serviço e a equipe(E, 2023).

O Bootstrap é um framework web com código-fonte aberto para desenvolvimento de componentes de interface e front-end para sites e aplicações web, usando HTML, CSS e JavaScript e tem como objetivo a construção de sites responsivos. O design responsivo garante que todos os elementos da interface funcionem seguindo o conceito mobile first, ou seja, que o design web inicialmente criado pensando em tablets e smartphones se adapte a outros dispositivos, como desktops(Ebac, 2023).

Node.js

O Node.js é um ambiente de execução do JavaScript no server side, que mostra a possibilidade de criar aplicações standalone em uma máquina servidor. Com o Node.js, utilizamos o gerenciador de pacotes NPM (Node Package Manager) sendo um diretório com um ou mais bibliotecas JavaScript, utilizado para inserir diversas funcionalidades a aplicativos ou scripts(Bessa, 2023). Foram utilizados frameworks para facilitar a utilização e melhora na eficiência do Node.js, sendo eles: Express, MySQL2, EJS, Nodemon, Sequelize.

A escolha se deve por necessitar de uma aplicação moderna, eficiente e de fácil entendimento, contribuindo para um sistema de qualidade para o usuário e possuindo uma coleta de dados sucinta.

MySQL

O MySQL Workbench é um sistema para design de banco de dados que integra design, desenvolvimento, criação e manutenção de SQL em um único ambiente de desenvolvimento. O MySQL Workbench possui um editor SQL que permite que você escreva e execute consultas SQL de forma eficiente, ele oferece vários recursos, facilitando assim a escrita e execução de consultas complexas(Andrade, 2020).

XAMPP

O XAMPP (Apache, MySQL, PHP, Pearl) é um software de desenvolvimento web amplamente empregada na indústria, simplificando a instalação e operação de um servidor web(Methaseo, 2023).

Inteligência Artificial

A Inteligência Artificial (IA) é um campo que estuda o desenvolvimento de sistemas capazes de realizar tarefas que normalmente requerem inteligência humana. A IA que será utilizada é Sistema de Recomendação.

O Sistema de Recomendação é um software que sugere conteúdos ou produtos com base no perfil do usuário e no comportamento anterior. Ele utiliza técnicas como filtragem colaborativa e análise preditiva para personalizar recomendações(Mensagem, 2023).

Sistemas Embarcados

Sistemas embarcados ou embutidos são sistemas computacionais especializados que atuam em conjunto com hardware e software, se responsabilizando por alguma função ou ação específica, no caso desse projeto, o sistema embarcado é representado pelo Arduíno(Souza, 2022).

O Arduino é uma plataforma de construção e prototipagem eletrônica que simplifica a criação de projetos interativos para o dia a dia, incluindo aplicações na Internet das Coisas(IoT). O Arduino é um hardware de código aberto, permitindo que qualquer pessoa possa aproveitar de suas funcionalidades. Com uma ampla gama de sensores disponíveis, o Arduino facilita a integração de diferentes componentes ao sistema(Thomsen, 2023). Se destaca o uso do arduino por conta de sua simplicidade e facilidade de programação, resistência e confiabilidade, compatibilidade com sensores e foco no controle de hardware.

O arduíno selecionado para ser o centro do sistema permitindo conexão wi-fi e coleta de dados é o UNO R4 WiFi, possuindo WiFi e Bluetooth integrados com ESP32-S3, matriz de LED 12x8 incorporada, periféricos avançados como barramento CAN, DAC de 12 bits e Amplificador Operacional

e retrocompatibilidade, pino para ligar/desligar de forma remota e tendo um microcontrolador Renesas RA4M1 com Arm Cortex-M4 de 32 bits.

Neste projeto, o hardware será composto por 3 tipos de sensores, sendo eles para monitoramento dos tanques, medidor de temperatura de água e medidor de pH e amônia presentes na água com ambos conectados a um arduino wi-fi para comunicação. Além de possuir um dispensador de ração que irá funcionar com horários e quantias de ração programadas e especificadas pelo usuário ao cadastrar o tanque no aplicativo.

O sensor de temperatura utilizado será o DS18B20, conhecido por sua alta precisão, utiliza o protocolo 1-Wire, o que permite a conexão de múltiplos sensores no mesmo barramento, facilitando a expansão do sistema para diferentes tanques.

Para a detecção de resíduos, será empregado o sensor MQ-135, sensível a diversos compostos tóxicos como amônia, dióxido de carbono e benzeno. Esse sensor oferece resposta rápida e boa estabilidade, contribuindo para a identificação precoce de alterações na qualidade da água.

Por fim, para a medição do pH, será utilizado o sensor pH0-14, possui uma sonda eletroquímica de vidro e é capaz de medir valores de pH entre 0 e 14, cobrindo toda a faixa relevante para ambientes aquáticos.

Aplicação IOT

O código foi desenvolvido para um ESP32 que monitora parâmetros ambientais de um cativeiro aquático, como temperatura, pH e amônia, por meio de sensores analógicos conectados aos pinos ADC1 do microcontrolador. As leituras são processadas e, no caso da temperatura, convertidas de voltagem para graus Celsius com base em uma fórmula específica. As informações são exibidas em tempo real em um display LCD I2C de 16 colunas por 2 linhas, permitindo o acompanhamento visual constante das condições do ambiente.

Além do monitoramento, o sistema inclui um mecanismo de alimentação automática controlado por um servo motor. A cada intervalo de tempo definido por padrão, a cada 1 minuto, o servo realiza um número de giros programado para simular a liberação de ração. Esses ciclos de alimentação podem ser ajustados dinamicamente pelo usuário através do monitor serial, oferecendo flexibilidade para diferentes necessidades de manejo.

Durante o funcionamento, o dispositivo exibe no LCD as leituras dos sensores e indica quando a alimentação está em andamento. Todas as ações são registradas no monitor serial, incluindo os valores coletados, o número de giros executados e mensagens de status. Dessa forma, o código fornece uma solução prática e automatizada para o controle de alimentação e monitoramento básico da qualidade da água em ambientes de criação.

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <ESP32Servo.h>

// Config LCD (ajuste o endereço se necessário, normalmente é 0x27 ou 0x3F)
LiquidCrystal_I2C lcd(0x27, 16, 2);

// Pinos dos sensores (ADC1 recomendado se usar WiFi)
const int amoniaPin = 32; // ADC1_4
const int phPin = 33;    // ADC1_5
```



```

const int tmpPin = 34;    // ADC1_6 (somente entrada)

// Pino do servo
const int servoPin = 18;

Servo myServo;

// Variáveis do sistema de alimentação
int ciclosDeRacao = 1;
unsigned long intervaloAlimentacao = 60000;
unsigned long ultimoAlimentacao = 0;

// Variáveis dos sensores
int AmoniaValue = 0;
int phValue = 0;
int tempValue = 0;

unsigned long ultimaLeitura = 0;
unsigned long intervaloLeitura = 2000;

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

    // Inicia LCD
    lcd.init();
    lcd.backlight();
    lcd.setCursor(0, 0);
    lcd.print("Sistema Iniciado");
    delay(2000);

    // Configura servo parado
    myServo.setPeriodHertz(50); // Frequência padrão para servos
    myServo.attach(servoPin);
    myServo.write(90);
    delay(1000);
    myServo.detach();
}

void loop() {
    // Leitura dos sensores
    if (millis() - ultimaLeitura >= intervaloLeitura) {
        AmoniaValue = analogRead(amoniaPin);
        phValue = analogRead(phPin);
        tempValue = analogRead(tmpPin);

        float voltage = tempValue * (3.3 / 4095.0); // ESP32 usa 3.3V e resolução de 12 bit
    }
}

```



```

float temperatureC = (voltage - 0.5) * 100;

lcd.clear();
lcd.setCursor(0, 0);
lcd.print("NH3: ");
lcd.print(AmoniaValue);
lcd.setCursor(0, 1);
lcd.print("pH: ");
lcd.print(phValue);
delay(500);

lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Temp: ");
lcd.print(temperatureC);
lcd.print(" C");
delay(500);

Serial.print("Amonia: ");
Serial.print(AmoniaValue);
Serial.print(" | pH: ");
Serial.print(phValue);
Serial.print(" | Temp: ");
Serial.print(temperatureC);
Serial.println(" C");

ultimaLeitura = millis();
}

if (millis() - ultimoAlimentacao >= intervaloAlimentacao) {
    alimentar();
    ultimoAlimentacao = millis();
}

if (Serial.available() > 0) {
    ciclosDeRacao = Serial.parseInt();
    if (ciclosDeRacao > 0) {
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print("Ciclos: ");
        lcd.print(ciclosDeRacao);
        Serial.print("Ciclos atualizados: ");
        Serial.println(ciclosDeRacao);
        delay(2000);
    }
}
}

```

```

}

void alimentar() {
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Alimentando...");
  Serial.println("Iniciando alimentacao");

  myServo.attach(servoPin);

  for (int i = 0; i < ciclosDeRacao; i++) {
    lcd.setCursor(0, 1);
    lcd.print("Giro: ");
    lcd.print(i + 1);
    Serial.print("Giro ");
    Serial.println(i + 1);

    for (int pos = 0; pos <= 180; pos++) {
      myServo.write(pos);
      delay(15);
    }

    delay(700);
    myServo.write(90);
    delay(100);
  }

  myServo.detach();

  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Alimentacao OK");
  Serial.println("Alimentacao concluida");
  delay(2000);
}

```

REFERÊNCIAS

ANDRADE, Ana Paula de. Principais aplicações para gerenciamento de banco de dados com interface gráfica. Treinaweb, 2020.

Disponível em: <https://www.treinaweb.com.br/blog/principais-aplicacoes-para-gerenciamento-de-banco-de-dados-com-interface-grafica>

.

AWARI. O papel do Diagrama Entidade-Relacionamento na modelagem de dados. Awari by Influence, 2023.

Disponível em: <https://awari.com.br/diagrama-entidade-relacionamento/>

.

BERTINI, Giovana et al. Camarões-de-Água-Doce do Vale do Ribeira: Riqueza, Importância Ecológica e Econômica, p. 71–72, 2021.

Disponível em: www.registro.unesp.br/Home/pesquisa5012/e-book_-manguezais_camaroes_manjuba_2021v090321.pdf

.

BESSA, André. Node.JS: o que é, como funciona esse ambiente de execução JavaScript e um Guia para iniciar. Alura, 2023.

Disponível em: <https://www.alura.com.br/artigos/node-js#:~:text=Runtime%20e%20V8-,0%20Node.,a%20ser%20executada%20pelo%20computador>

.

BLAVA, Jônatan de Oliveira. A metodologia Canvas e suas variações para o desenvolvimento do empreendedorismo, 2017.

Disponível em: <http://repositorio.unesc.net/bitstream/1/5617/1/J%C3%B4natan%20de%20Oliveira%20Blava.pdf>

.

DANTAS, Matheus Fernandes do Nascimento. Um sistema de telemonitoramento e automação baseado em rede LoRa para criação de camarão. Universidade Federal do Ceará - Campus de Quixadá, Quixadá, CE, p. 16–48, 2020.

Disponível em: <http://www.repositorio.ufc.br/handle/riufc/61231>

.

E, Carlos. O Que é JavaScript e Para Que Serve na Programação Web. Hostinger, 2023.

Disponível em: <https://www.hostinger.com.br/tutoriais/o-que-e-javascript>

.

EBAC. O que é Bootstrap e como utilizá-lo. Escola Britânica de Artes Criativas & Tecnologia, 2023.

Disponível em: <https://ebaonline.com.br/blog/o-que-e-bootstrap>

.

G, Ariane. O que é CSS? Guia Básico para Iniciantes. Hostinger, 2022.

Disponível em:

<https://www.hostinger.com.br/tutoriais/o-que-e-css-guia-basico-de-css>

.

LONGEN, Andrei Silveira. O Que é HTML: O Guia Definitivo para Iniciantes. Hostinger, 2023.

Disponível em:

<https://www.hostinger.com.br/tutoriais/o-que-e-html-conceitos-basicos>

.

LOPES, Michele. O que é Figma e como usar? Escola Britânica de Artes Criativas & Tecnologia, 2023.

Disponível em: <https://ebaonline.com.br/blog/o-que-e-figma-e-como-usar>

.

LUCIDCHART. Diagrama de caso de uso UML: O que é, como fazer e exemplos. LucidChart, 2024a.

Disponível em: <https://www.lucidchart.com/pages/pt/diagrama-de-caso-de-uso-uml#:~:text=um%20diagrama%20UML-,0%20que%20%C3%A9%20diagrama%20de%20caso%20de%20uso%3F,de%20s%C3%ADmbolos%20e%20conectores%20especializados.>

.

LUCIDCHART. O que é o Diagrama UML. LucidChart, 2024b.

Disponível em: <https://www.lucidchart.com/pages/pt/o-que-e-uml#:~:text=De%20modo%20geral%2C%20diagramas%20UML,por%20meio%20de%20diagramas%20UML>

.

LUCIDCHART. O que é um diagrama de classe UML? LucidChart, 2024c.

Disponível em:

<https://www.lucidchart.com/pages/pt/o-que-e-diagrama-de-classe-uml>

.

LUCIDCHART. O que é um diagrama de objetos? LucidChart, 2024d.

Disponível em:

<https://www.lucidchart.com/pages/pt/o-que-e-diagrama-de-objetos-uml>

.

MENSAGEM, Meios. Algoritmo de recomendação: o que é e como ele contribui para as estratégias personalizadas. meio & mensagem, 2023.

Disponível em: <https://www.meioemensagem.com.br/proxima/pxx-noticias/algoritmo-de-recomendacao>

.

METHASEO. O que é: XAMPP (Cross-Platform, Apache, MySQL, PHP, and Perl). Methaseo, 2023.

Disponível em: <https://methaseo.com.br/glossario/o-que-e-xampp-cross-platform-apache-mysql-php-and-perl/>

.

ROCHA, Diego Maia. Produção, Mercado e Promoção do consumo de camarão: Por que aumenta, dia a dia, a necessidade de entendermos um pouco mais sobre o que anda acontecendo com cada um desses temas? Aquaculture Brasil, 2023.

Disponível em:

<https://www.aquaculturebrasil.com/coluna/369/producao,-mercado-e-promocao-do-consumo-de-camarao:-por-que-aumenta,-dia-a-dia,-a-necessidade-de-entendermos-um-pouco-mais-sobre-o-que-anda-acontecendo-com-cada-um-desses-temas>

.

SOUZA, Fábio. O que são sistemas embarcados? Embarcados, 2022.

Disponível em: <https://embarcados.com.br/o-que-sao-sistemas-embarcados/>

.

THE GLOBAL GOALS. The 17 Goals.

Disponível em:

<https://www.globalgoals.org/goals/12-responsible-consumption-and-production/>

.

THOMSEN, Adilson. O que é Arduino, para que serve e primeiros passos. MakerHero, 2023.

Disponível em: <https://www.makerhero.com/blog/o-que-e-arduino/>

.

UDDIN, Mohammad Salah et al. Freshwater shrimp farm monitoring system for Bangladesh based on internet of things. **Engineering Reports**, v. 2, n. 7, p. 2–12, 2020.

Disponível em: <https://doi.org/10.1002/eng2.12184>

.

ZAINUDDIN, Zaryanti; IDRIS, Riswan; AZIS, Asmawaty. Water Quality Monitoring System for Vannamae Shrimp Cultivation Based on Wireless Sensor Network In Taipa, Mappakasunggu District, Takalar. Atlantis Press, p. 89–92, 2019.

Disponível em: <https://doi.org/10.2991/icmeme-18.2019.20>

.