



AUTOMATIZAÇÃO DE AQUÁRIO PARA REPRODUÇÃO DE BETTA SPLENDENS EM CATIVEIRO

Palavras-Chave: Automatização, Sistemas Embarcados, Aquicultura

Guilherme Felipe De Campos - Cotuca - Unicamp, João Guilherme Pereira dos Santos - Cotuca - Unicamp, João Pedro Ferreira Barbosa - Cotuca - Unicamp, Nicolas Militão Livotto - Cotuca - Unicamp
Prof. Me. Sergio Luiz Moral Marques - Cotuca - Unicamp

INTRODUÇÃO

O cultivo e criação de peixes ornamentais, em especial o *Betta splendens*, têm ganhado destaque nas últimas décadas, impulsionado pela crescente demanda do mercado de aquarismo. No entanto, a reprodução dessa espécie em cativeiro apresenta desafios que podem afetar a eficiência e o bem-estar dos exemplares.

Nesse contexto, a automação de aquários surge como uma abordagem promissora para otimizar as condições ambientais e alcançar parâmetros ideais que favoreçam a reprodução bem-sucedida. Este trabalho tem como objetivo o desenvolvimento de um sistema completo de automatização de aquários para a reprodução de *Betta splendens* em cativeiro. Para isso, utilizamos dispositivos eletrônicos como Arduino Uno, ESP32 e ESP32-CAM, bem como linguagens de programação C e MicroPython para prototipagem.

A partir do hardware concebido, monitoramos e controlamos de forma remota os parâmetros ambientais críticos, como temperatura, pH da água e o fotoperíodo, essenciais para o processo reprodutivo dos peixes. Além disso, foram desenvolvidos um website e um serviço web — API REST e servidor WebSocket — para a visualização e transmissão em tempo real das imagens das câmeras e dos dados dos sensores, proporcionando uma plataforma interativa e eficiente para o acompanhamento contínuo do sistema, permitindo aos pesquisadores realizarem análises detalhadas e tomarem decisões baseadas em informações atualizadas.

METODOLOGIA

A metodologia adotada para alcançar os objetivos propostos envolveu três etapas:

1. Discussão e automação da iluminação

Inicialmente, realizamos um estudo detalhado sobre as necessidades e preferências reprodutivas do *Betta splendens* em cativeiro em consonância com outra pesquisa, também conduzida por nosso coorientador, Prof. Dr. Jodir Pereira da Silva, intitulada “Estudo da Influência de Parâmetros Ambientais no Comportamento Reprodutivo de *Betta splendens* em Cativeiro”. Inicialmente, priorizou-se a automatização da iluminação dos aquários, definindo o fotoperíodo — exposição à luz durante um dia para a amostra —, sem que houvesse a necessidade de um pesquisador comparecer presencialmente ao laboratório e manualmente acender e apagar as luzes no fotoperíodo de interesse. Para tanto, utilizou-se um Arduino Uno, um sensor de luminosidade — LDR, Light Dependent Resistor — e um relé, além de outros componentes eletrônicos acessórios. Uma vez prototipado, a iluminação ficou dependente do controle do sistema idealizado, como pode ser verificado na Figura 1.

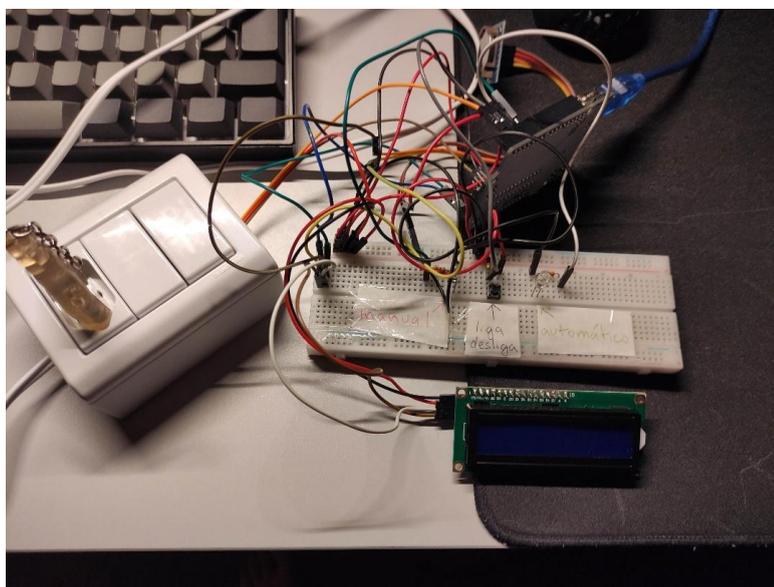


Figura 1 — Circuito para automatização da iluminação

2. Coleta e visualização dos parâmetros

Após a rápida prototipagem do sistema de iluminação, tivemos maior liberdade foi concebida uma arquitetura geral do funcionamento dos demais componentes a serem incorporados ao sistema, levando em consideração os requisitos funcionais dos pesquisadores do outro referido projeto de pesquisa, que foram:

- Monitoramento remoto da temperatura e pH da água e fotoperíodo dos aquários;
- Transmissão ao vivo dos aquários;
- Gravação de períodos pré-determinados e upload em plataforma de vídeos na nuvem e
- Website e/ou aplicativo para acesso aos parâmetros.

A arquitetura inicial resultante pode ser verificada na Figura 2.

Arquitetura para automação dos aquários

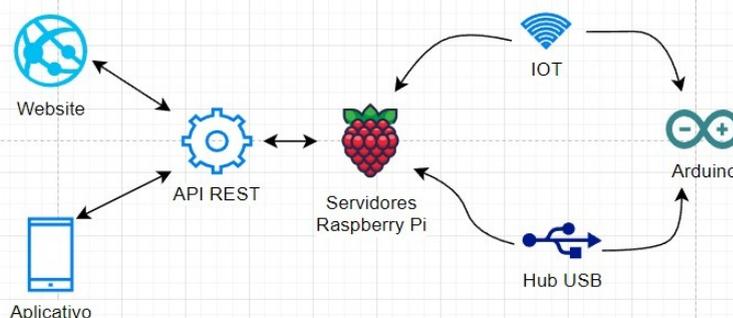


Figura 2 – Arquitetura geral inicial

3. Prototipagem e desenvolvimento do sistema completo

Tendo uma vez estabelecido como o sistema funcionaria de forma geral, partimos para o desenvolvimento do serviço web “cerne” – Core Web Service –, que conta com uma API REST – Application Programming Interface e Representational State Transfer – e servidor WebSocket para armazenamento definitivo e transmissão de dados em tempo real, respectivamente.

Com orientação “*backend first*”, somente após a conclusão do desenvolvimento do que roda por trás dos panos – a referida API REST – é que desenvolvemos o *frontend* para acesso às transmissões e dados dos aquários.

Posteriormente ao desenvolvimento do *backend* e *frontend* do projeto, prototipamos todo o hardware necessário, incluindo sensores de luminosidade, temperatura, sondas de pH e câmeras, além dos demais componentes eletrônicos acessórios, como *protoboards*, cabos *jumper* e variados resistores, que estaria conectada à internet – ESP32 –, diferentemente do sistema independente e redundante de iluminação, que não tem conexão à internet – Arduino Uno.

Com as câmeras – ESP32-CAM – devidamente programadas, instaladas e configuradas, como último software a ser desenvolvido, programamos um *worker* que observa as imagens em tempo real e, após gravar os aquários durante um período de interesse dos pesquisadores, realiza o *upload* dos vídeos para um serviço externo de hospedagem de vídeos na nuvem.

RESULTADOS E DISCUSSÃO

Conseguimos desenvolver e prototipar o que havíamos proposto inicialmente em consonância com a outra referida pesquisa, levando em consideração o que fora estabelecido na metodologia do projeto, concluindo todos os escopos: dados de fotoperíodo, temperatura e pH, transmissão ao vivo e gravação dos aquários disponibilizados através de um website, que pode ser visitado através do endereço <<https://aquadynamics.vercel.app/>>. O diagrama geral inicial proposto sofreu algumas variações, uma vez que foram tomadas decisões de engenharia que divergiram do que havia se prospectado. O diagrama final pode ser verificado na Figura 3.

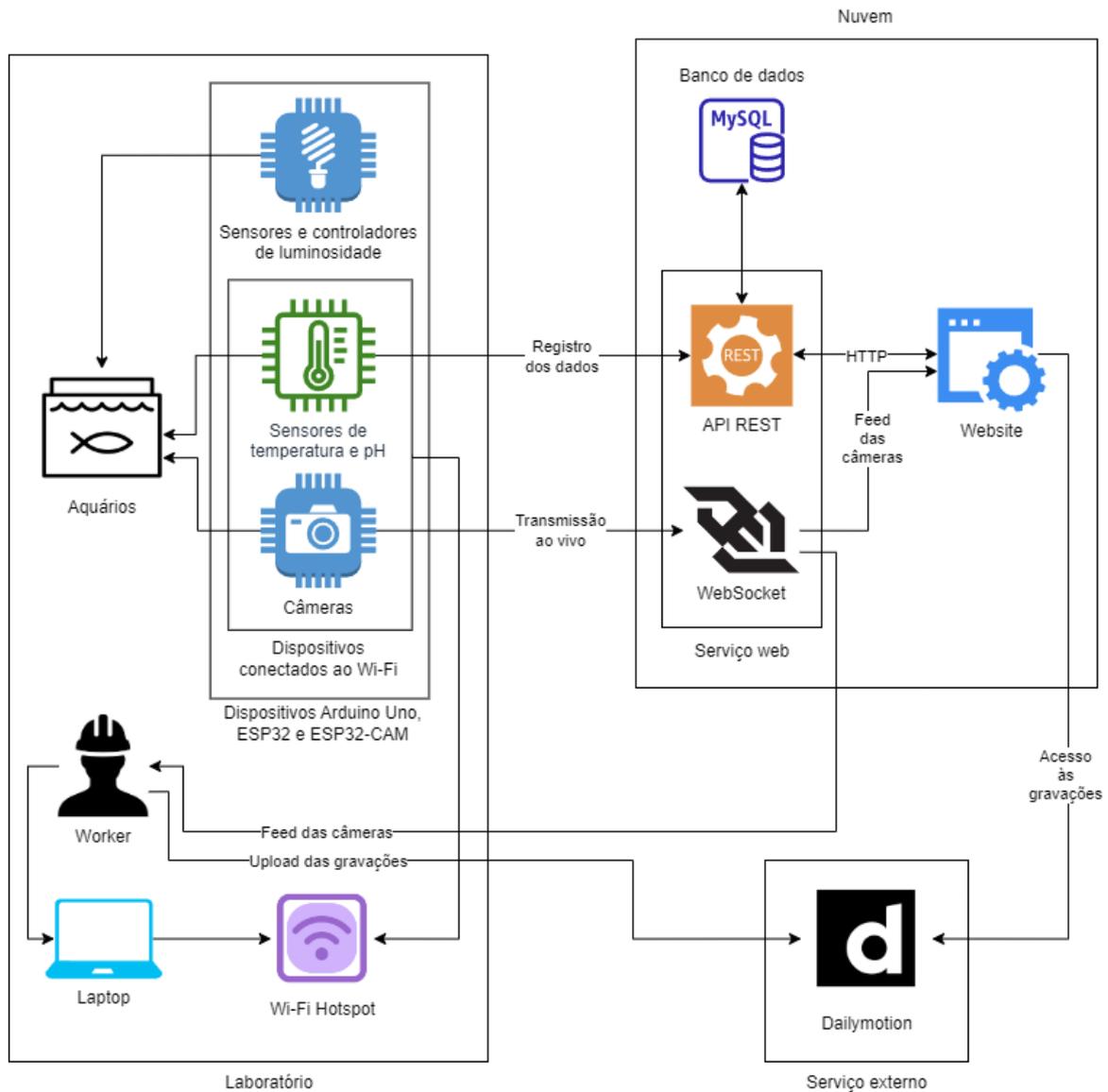


Figura 3 — Arquitetura geral final

Optamos por utilizar o serviço Dailymotion para *upload* das gravações por sua flexibilidade e facilidade de integração. Além da programação dos microcontroladores em linguagem Arduino, C, C++ e MicroPython, utilizamos Python para o *worker* e TypeScript com Express.js e Node.js, aplicando conceitos de *Clean Architecture*, princípios S.O.L.I.D. e *Domain Driven Design* para o *backend* e TypeScript com Next.js e React.js para o *frontend*, que se provaram ótimas ferramentas para desenvolvimento *full stack*. O código necessário para a realização do projeto inteiro está agregado e disponibilizado no repositório <<https://github.com/ojpbarbosa/aquadynamics>>.

CONCLUSÕES

Concluimos que a automação de aquários para a reprodução de *Betta splendens* em cativeiro, por meio do uso de dispositivos como Arduino Uno, ESP32 e ESP32-CAM, demonstrou-se uma abordagem eficaz para estabelecer parâmetros ideais e favorecer o sucesso reprodutivo dessa espécie.

O desenvolvimento de um website completo com API e backend, associado à transmissão em tempo real das imagens das câmeras e dos dados dos sensores, possibilitou um acompanhamento contínuo e eficiente dos aquários de reprodução, permitindo intervenções rápidas e precisas para garantir o bem-estar dos peixes e a eficiência do processo reprodutivo.

Esperamos que os resultados obtidos com este estudo contribuam para o aprimoramento das práticas de criação de *Betta splendens* em cativeiro e também inspirem futuras pesquisas no campo da aquicultura, utilizando a automação como uma ferramenta poderosa para o avanço científico e para a conservação dessas espécies tão apreciadas no mercado de aquarismo. O sistema desenvolvido abre novas perspectivas para o aperfeiçoamento das técnicas de reprodução em cativeiro, contribuindo para a conservação da biodiversidade e o desenvolvimento sustentável da aquicultura.

BIBLIOGRAFIA

- SANTOS, Sara; SANTOS, Rui. **Getting Started with the ESP32 Development Board**. Random Nerd Tutorials. Disponível em: <<https://randomnerdtutorials.com/getting-started-with-esp32/>>
- SANTOS, Sara; SANTOS, Rui. **Getting Started with MicroPython on ESP32 and ESP8266**. Random Nerd Tutorials. Disponível em: <<https://randomnerdtutorials.com/getting-started-micropython-esp32-esp8266/>>.
- SANTOS, Sara; SANTOS, Rui. **ESP32 DS18B20 Temperature Sensor with Arduino IDE (Single, Multiple, Web Server)**. Random Nerd Tutorials. Disponível em: <<https://randomnerdtutorials.com/esp32-ds18b20-temperature-arduino-ide/>>.
- SANTOS, Sara; SANTOS, Rui. **Upload Code to ESP32-CAM using ESP32-CAM-MB USB**. Random Nerd Tutorials. Disponível em: <<https://randomnerdtutorials.com/upload-code-esp32-cam-mb-usb/>>.
- CASS, Stephen. **Top Programming Languages 2022 > Python's still No. 1, but employers love to see SQL skills**. IEEE Spectrum, 23 ago. 2022. Disponível em: <<https://spectrum.ieee.org/top-programming-languages-2022>>.
- MARTIN, Robert Cecil. **Clean Architecture: A Craftsman's Guide to Software Structure and Design**. Pearson, 12 set. 2017.