

# Relatório Parcial

Título do Projeto: Boia para Monitoramento e  
Prevenção de Inundações

ID JEMS: 242989

Estudantes: José Batista de Souza Júnior, Larissa da Silva Matos e  
Samuel Henrique Guimarães Alencar.

Professor: Wagner Guimarães Al-Alam

Universidade: Universidade Federal do Ceará (UFC)

## Declaração de Responsabilidade

Nós, estudantes do projeto ID. 242989, declaramos que as informações contidas neste relatório são verdadeiras, sob a supervisão do nosso professor orientador que assina abaixo.

Assinatura do Professor: \_\_\_\_\_

Nome do Professor: Wagner Guimarães Al-Alam

Data: 28/10/2024

## Cronograma Inicial Proposto

Início - Fim	Etapa
08/07 - 22/07	Compra de componentes e definição do modelo 3D da boia
23/07 - 30/09	Impressão do modelo 3D e desenvolvimento do firmware
24/08 - 01/09	Montagem do sistema embarcado e modelo 3D
02/09 - 30/09	Desenvolvimento da API para disponibilização dos dados
01/10 - 14/10	Escrita do relatório parcial
15/10 - 03/11	Testes, validações e ajustes
04/11 - 18/11	Escrita do relatório final

## Cronograma Atualizado

Início - Fim	Etapa	Status (%)
30/06	Envio da Proposta	100%
05/08 - 11/08	Embasamento teórico	100%
05/08 - 11/08	Definição da arquitetura da solução	100%
05/08 - 11/08	Definição da metodologia de desenvolvimento	100%
12/08 - 18/08	Definição das linguagens	100%
12/08 - 18/08	Compra dos sensores	100%
12/08 - 18/11	Modelagem dos tempos de resposta	50%
19/08 - 01/09	Implementação das interfaces da arquitetura	100%
19/08 - 24/10	Desenvolvimento do protótipo	100%
26/08 - 18/11	Modelagem 3D	20%
26/08 - 18/11	Desenvolvimento de teste de conectividade e funcionamento LoRa	20%
02/09 - 08/09	Desenvolvimento do driver do acelerômetro	100%
02/09 - 15/09	Desenvolvimento do driver do sensor de temperatura	100%
02/09 - 22/09	Desenvolvimento do driver do módulo LoRa	100%
16/09 - 18/11	Estudo e Implementação da interface de comunicação com a rede Helium	10%
16/09 - 06/09	Implementação da comunicação via LoRa	100%
23/09 - 28/10	Elaboração do relatório parcial	100%
28/10	Submissão do Relatório Parcial	100%
18/11	Submissão do Relatório Final	0%
26/11 – 29/11	Apresentação no SBESC	0%

## Dificuldades e Soluções

Durante o desenvolvimento do projeto, foram encontradas algumas dificuldades que impactaram as decisões de projeto.

Uma das primeiras dificuldades foi a concepção inicial da boia, que parecia muito suscetível a erros, devido às variáveis externas como a sujeira, luz, calor e interferências do ambiente aquático. Para contornar esse problema, foi decidido simplificar o mecanismo de medição, substituindo a boia por um corpo flutuante que ficaria dentro de um cano fixado ao chão. Esse método apresentou eficácia por ser simples, leve e funcional, além de minimizar as chances de erro na medição, já que a bolinha opera dentro de um ambiente mais controlado: o interior do cano.

Outro desafio foi a escolha dos sensores mais adequados. Inicialmente foi discutido o uso de sensores infravermelhos e ultrassônicos, mas ambos apresentavam limitações, especialmente no que diz respeito à precisão e suscetibilidade a interferências externas. Após uma análise mais aprofundada, decidimos optar pelo sensor laser [1], que se mostrou mais preciso e confiável para medir o nível da água. Alguns fatores que foram levados em consideração para escolha desse sensor específico foram: precisão, frequência de leitura, distância de medição, preço, consumo de energia e protocolo de comunicação.

Além disso, foi priorizado o protocolo de comunicação **UART** em vez de **I2C**, devido à sua menor complexidade, o que facilita a integração com a placa e reduz a necessidade de gerenciamento adicional de múltiplos dispositivos no mesmo barramento.

Também foram enfrentadas dificuldades na configuração e inicialização do módulo LoRa. A falta de exemplos claros em relação ao uso das duas UARTs disponíveis no módulo, bem como o envio de mensagens de um nó para outro, dificultou o processo. Foram testadas diferentes configurações até encontrar a correta para o envio de mensagens entre dispositivos, garantindo que a comunicação entre a boia e o sistema remoto funcionasse apropriadamente. Para solucionar essas questões, foi necessário seguir o tutorial **LoRa Mesh da Radioenge** [2], para configurar corretamente o sistema. Além disso, para a parte de comunicação entre dois dispositivos utilizando *strings*, foi utilizado o repositório **LoRaMESH - ESP8266** [3].

Outra adversidade encontrada foi a integração com a **rede Helium** [4]. Devido à ausência de cobertura da rede Helium na região, não foi possível implementar o sistema utilizando essa rede no protótipo. No entanto, a proposta mantém a integração com essa rede, considerando a possibilidade de futura expansão da cobertura ou ativação de *gateways* nas proximidades.

Outra dificuldade encontrada foi que a rede Helium atualmente oferece suporte exclusivo ao padrão **LoRaWAN** [4]. Essa tecnologia utiliza uma arquitetura de comunicação em estrela, onde os dispositivos se comunicam diretamente com *gateways* LoRaWAN, em vez de se comunicarem entre si, como nas redes **LoRaMesh**, utilizadas pelos módulos recebidos. Devido a essa incompatibilidade, foi proposto o desenvolvimento de um nó *bridge* para resolver o problema. Esse nó seria responsável por receber as mensagens LoRaMesh, convertê-las para o padrão LoRaWAN, e enviá-las para o *gateway* LoRaWAN. No entanto, essa solução ainda não foi implementada.

Outro obstáculo identificado foi o plano inicial de desenvolver uma API própria para o acesso aos dados. Devido à alta demanda de tempo para essa tarefa, foi escolhida uma solução mais rápida: a utilização do **Node-RED**. Essa ferramenta permite uma visualização e interação mais ágil com os dados. O sistema, então, foi organizado da seguinte forma: um **nó transmissor**, que coleta os dados dos sensores na boia, e um **nó receptor**. A solução envolve o uso de uma conexão **MQTT**, em que o **broker** interage com o Node-RED, permitindo que o usuário acesse e visualize as mensagens recebidas por meio de uma interface gráfica.

Outra dificuldade enfrentada durante o desenvolvimento do projeto foi a seleção de um **protocolo de monitoramento** eficiente para garantir a continuidade da transmissão de dados em caso de falhas de comunicação. Dessa maneira, optou-se pelo uso do protocolo **Heartbeat** [5] para monitorar a integridade da comunicação. O Heartbeat permite que o sistema identifique falhas no envio ou recebimento de dados e tome medidas corretivas.

Outra complicação enfrentada foi relacionada ao campo de visão do sensor *laser*. Ele possui cones de atuação, e seu alcance é limitado pelo ponto onde esses cones tocam as paredes internas do cano, o que reduz a precisão em medir a distância total. De acordo com os cálculos realizados, o alcance efetivo do sensor é de **225,54 mm**, valor que foi confirmado nos testes práticos com o sensor dentro do cano de **50 mm de diâmetro e 1m de altura**. A solução adotada no momento foi utilizar um cano com diâmetro de **100 mm e 95 cm de altura**, o que aumentou a área de atuação do sensor, permitindo cobrir uma faixa maior de flutuação do corpo flutuante. No entanto, essa solução não permitiu cobrir toda a altura do cano, pois, em certo ponto, o sensor começa a detectar as paredes internas. Assim, para o cano de 100 mm, o alcance útil estimado foi de **451,07 mm**.

Diante do custo elevado e plena necessidade de uso do painel solar e bateria recarregável apenas na versão de produto final, optou-se por utilizar um *power bank* como solução temporária para o protótipo. Como solução temporária, optou-se por

utilizar um **power bank de 5000mAh**, que atende adequadamente às necessidades energéticas do protótipo. Essa solução permite continuar o desenvolvimento apenas com as interrupções de carregamento do *power bank* enquanto ainda serão feitos cálculos detalhados de **autonomia**, levando em consideração o consumo médio do sistema. Esses cálculos ajudarão a estimar o tempo de funcionamento da boia com a bateria utilizada, garantindo que o sistema continue funcional por um período razoável antes de necessitar de recarga.

## Referências

- [1] SHENZHEN HCJ TECHNOLOGY CO., LTD. **TOF10120 Time-of-Flight Ranging Sensor**. Datasheet, 2024. Disponível em: <https://www.electronicclinic.com/wp-content/uploads/2021/06/TOF10120-Datasheet.pdf>. Acesso em: 06 out. 2024.
- [2] CEREZA, Gustavo. **LoRa Mesh da Radioenge Tutorial Completo**. Elcereza - Faça a sua inovação, 2022. Disponível em: <https://elcereza.com/lora-mesh-da-radioenge-tutorial-completo/>. Acesso em: 25 set. 2024.
- [3] RADIOENGE. **LoRaMESH - ESP8266**. Github, 2020. Disponível em: <https://github.com/Radioenge/LoRaMESH-ESP8266/tree/master>. Acesso em: 25/09/2024.
- [4] HELIUM SYSTEMS INC. **Helium – introducing the people’s network**. Helium, 2024. Disponível em: <https://www.helium.com/>. Acesso em: 06 out. 2024.
- [5] TECNOLOGIA, C. **O que é: Heartbeat**. Credited Tecnologia. Disponível em: <https://tecnologia.credited.com.br/glossario/o-que-e-heartbeat-entenda-seu-funcionamento/>. Acesso em: 06 out. 2024.
- [6] RADIOENGE; CEREZA, G.; CHIODI M.; TABAROSKI G. **LoRa Mesh da Radioenge**. Github, 2019. Disponível em: <https://github.com/Radioenge/LoRaMESH>. Acesso em: 25 set. 2024.
- [7] NCDCOMMUNITY. **Interfacing SHT30 with ESP32**. Github, 2019. Disponível em: <https://github.com/ncdcommunity/Esp32-And-SHT30-temperature-and-humidity-sensor/tree/master>. Acesso em: 05 out. 2024.
- [8] SENSIRION. **Datasheet SHT3x-DIS - Humidity and Temperature Sensor**. Sensirion, 2016. Disponível em: [https://www.mouser.com/datasheet/2/682/Sensirion\\_Humidity\\_Sensors\\_SHT3x\\_Datasheet\\_digital-971521.pdf](https://www.mouser.com/datasheet/2/682/Sensirion_Humidity_Sensors_SHT3x_Datasheet_digital-971521.pdf). Acesso em: 05 out. 2024.
- [9] SCHORCHT, Gunar. **Driver for SHT3x digital temperature and humidity sensor**. Github, 2017. Disponível em: <https://github.com/gschorcht/sht3x-esp-idf/tree/master>. Acesso em: 06 out. 2024.
- [10] DEWMITH, Sarathee. **SHT3X-DIS Arduino Library**. Github, 2023. Disponível em: <https://github.com/barbarossa12/sht3x-dis-arduino-lib/tree/main>. Acesso em: 06 out. 2024.