

Se²Oil – Satélite de monitoramento de manchas de óleo em rios, lagos e lagoas

Ana Laura de Matos Placidino

Daniel de faveri Toledo

Gabriel Yago Alves Mendonça

Pedro Juan Leme Azeredo

Alan Barbosa de Paiva (orientador)

2023

Resumo

Todos os dias pessoas despejam 200 milhões de litros de óleo de cozinha o que contamina mais de 5 trilhões de litros de água doce em todo mundo, bloqueando o processo de fotossíntese e prejudicando os seres vivos causando sua morte ou algum tipo mutação, além de ser um ato contra a lei.

Pensando nesse problema, decidimos montar um satélite que irá identificar as manchas de óleo em rios. Isso ocorrerá por meio de uma câmera infravermelha, que detectará essas manchas e Enviará os dados, analisados por meio de um algoritmo de KNN, para a estação de telemetria em terra.

Palavras-chave: Contaminação. Óleo. Água. Satélite.

1. Introdução

Anualmente, o Brasil produz cerca de 9 bilhões de litros de óleo de cozinha (ECOLEO,2023). Segundo a *Oil World*, mais de 200 milhões de litros desse mesmo óleo vão para rios e oceanos, o que só nos mostra a urgência que se deve ter para prevenir toda essa contaminação (OILWORLD,2023). O óleo de cozinha, se for descartado de maneira incorreta, traz inúmeras consequências negativas e prejudiciais ao meio ambiente como por exemplo o descontrole do oxigênio, e a morte de peixes e seres vivos que ali habitam. Segundo dados da SABESP, 1 litro de óleo pode contaminar até 25 mil litros de água, quando todo esse óleo chega nos cursos de água (SABESP,2023).

Vamos separar o petróleo do óleo porque, mesmo ambos sendo muito parecidos, seus danos ao meio ambiente serão diferentes. O vazamento por petróleo nos oceanos ocorre sempre por intermédio de defeitos em navios-petroleiros ou rompimento de algum duto. O primeiro problema causado por esse tipo de vazamento é que ele é escuro e, ao entrar em contato com a água, atrapalha a entrada de luz, prejudicando assim o fitoplâncton que fica na superfície da água impedindo que ele realize fotossíntese, alterando e prejudicando assim os seres vivos e a cadeia alimentar, atingindo os seres humanos que se alimentam de algum animal que teve contato com o óleo contaminado.

Outro problema identificado após esse vazamento é que os animais que tiverem o contato com esse óleo podem morrer e/ou sofrer mutações em seu DNA, ter seu sistema nervoso atingido. As aves atingidas podem sofrer com o desequilíbrio térmico, morrendo de frio ou de calor (MORAES, 2022).

Quando comparamos o volume dos vazamentos, os derramamentos de petróleo acontecem poucas vezes e seu volume é bem grande, já a contaminação de água por meio do óleo de cozinha acontece todos os dias, em escalas pequenas mas de forma mais distribuída. Segundo a ABIOVE (Associação Brasileira da Indústria de Óleos Vegetais) com o IBGE (Instituto Brasileiro de Geografia e Estatísticas) os brasileiros, por ano, despejam cerca de 1 bilhão de litros de óleo incorretamente (RECICLA SAMPA,2023)

Em relação aos vazamentos decorrentes pelo óleo de cozinha podemos citar o caso de 2000, na Baía de Guanabara, em que cerca de 1.300 m³ de óleo caíram no mar. As equipes até tentaram remover aquele óleo que causou a morte e mutações genéticas de vários seres vivos (JÚNIOR, 2018).

Em 1972, depois da conferência de Estocolmo, O Brasil criou um órgão chamado SEMA – Secretaria Especial do Meio Ambiente que tem o objetivo de garantir a implementação de políticas no meio ambiente e que, essas políticas, possam conservar os recursos naturais, auxiliando assim na qualidade de vida e no desenvolvimento sustentável (JÚNIOR, 2018).

Segundo a Lei nº 14.69 foi proibido o descarte de óleo no meio ambiente. Sendo assim, é contra a lei o despejo desse produto no meio ambiente, já que o mesmo causa inúmeras doenças por descarte inadequado e, ao entrar em contato com pragas e bactérias que podem matar e/ou causar doenças tanto para pessoas como para animais, isso porque no óleo de cozinha usado há alguns componentes químicos tóxicos a saúde (MORAES, 2022)

Um deles é benzeno ele pode causar doenças como leucemia, anemia e distúrbios de comportamento, isso ocorre porque ele entra na medula óssea e nos tecidos gordurosos. Outra substância tóxica é o tolueno, as pessoas que ingerirem sofrem de convulsões no mesmo instante que ingerida, sufocamento ou até mesmo a morte por parada cardíaca. Esse componente envenena o sistema nervoso central impedindo que este trabalhe normalmente. Outro componente que é tóxico é o xileno, causando irritação na pele, nos olhos e respiratório, causando tonturas e asfixia, ele também mexe com o sistema nervoso central (MORAES, 2022)

2. Objetivo da missão

A missão em questão visa o desenvolvimento de um satélite equipado com uma câmera infravermelha capaz de detectar e identificar manchas de óleo na superfície de corpos d'água, como rios e lagos. A

detecção precoce dessas manchas é de suma importância para a proteção e conservação dos cursos de água, bem como para a sobrevivência da fauna e flora que neles habitam, além de garantir a segurança e saúde das comunidades humanas que dependem desses recursos hídricos. A tecnologia proposta busca, portanto, contribuir de forma significativa para a gestão ambiental e para a promoção da sustentabilidade em nosso planeta.

3. Funções e responsabilidades da equipe

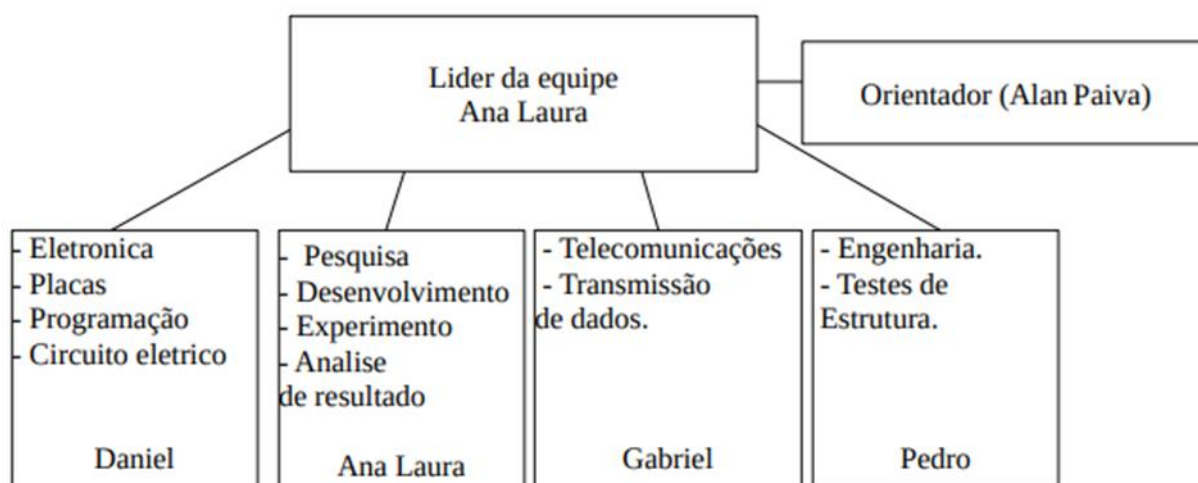


Figura 1 – Organograma da equipe
Fonte: autoria propria

4. Projeto conceitual

Numa primeira fase do projeto, com base na diferença entre o calor específico da água e do óleo, ou seja, na capacidade de absorver calor de ambos, chegamos a conclusão que a água e o óleo devem ter uma diferença de temperatura enquanto estiverem mesmo sobe a luz solar, mesmo sabendo dos princípios de equilíbrio térmico de dois líquidos em contato. Para comprovar esta teoria, realizamos o experimento descrito abaixo.

Tabela 1 – Características físico-químicas da água, óleo de soja, e óleo combustível

	ÁGUA	ÓLEO DE SOJA	ÓLEO COMBUSTÍVEL
Calor Específico	1 cal/g.°C	0.47 kcal/kg	0,4 a 0,5 kcal/kg
Ponto de ebulição	100° C	226 °C	180 a 360°C



Figura 2: Experimento de medição de temperatura utilizando termômetro infravermelho.

Fonte: Autoria própria.

Materiais:

- Termômetro Infravermelho AD801.
- 700ml de água em temperatura ambiente.
- 30ml de óleo
- Béquer de plástico de 1 litro

Objetivo:

Determinar se o sensor infravermelho consegue detectar se há diferença de temperatura entre a água e o óleo

Descrição do experimento:

Neste experimento adicionamos 700 mL em um béquer de 1 litro e medimos a temperatura da água e do óleo separados. Após esse teste, adicionamos os 30 mL de óleo de soja na superfície da água, sem contato com o Sol e medimos a temperatura da água e da mancha de óleo. Depois colocamos o béquer num local iluminado pelo Sol e aguardamos 10 minutos para fazer a nova leitura com o sensor, da água e do óleo novamente. Todas as leituras foram repetidas 3 vezes e depois tiramos a média simples dos valores. As medições foram feitas com uma distância aproximada de 30 cm, utilizando o modo “superfície” do termômetro.

Tabela 2-Experimento comparativo de temperatura do óleo e da água

TEMPERATURA AMBIENTE		Repetições	Água e óleo misturados na sombra		Água e óleo misturados no Sol.	
			--		10 min.	
			Água	Óleo	Água	Óleo
Óleo	Água	TESTE 01	24,2	24,5	26,3	26,5
25,7	24,2	TESTE 02	24,2	24,6	26,0	26,5
		TESTE 03	24,6	24,6	26,2	26,5
		Média	24,3	24,5	26,1	26,5

Com base nos resultados do experimento, determinamos que a temperatura medida do óleo tem uma diferença mínimo de 0,2 °C em relação a temperatura da água e que, essa diferença pode chegar até 0,4°C.

5. Detalhes operacionais

5.1. Detalhes dos subsistemas

Para facilitar o trabalho do desenvolvimento do satélite, os sistemas do satélite foram divididos em cinco sistemas:

- O sistema de alimentação e medição da bateria é responsável pela parte de alimentação do sistema eletrônico do satélite. O sensor NTC-10K monitora a temperatura da bateria, desligando caso a temperatura atinja um valor superior a 60 °C. Um divisor de tensão fornece a quantidade de energia disponível na bateria, que será enviado no pacote de dados do satélite. No projeto ainda deve ser adicionado um subsistema de recarga, usando placas solares de 1W, para recarregar o sistema elétrico do satélite. Pretendemos usar duas baterias Li-Ion 18650 2,2 A, com taxa de descarga de 2C.
- O sistema de telemetria será o conjunto de sensores para monitoramento da posição do satélite, tanto em relação a sua posição geográfica quanto a sua posição. Serão quatro sensores: Módulo GPS GY-NEO6MV2, que ficará responsável pela medição geográfica do satélite; Barômetro BMP180 responsável por medir a altitude do satélite; Módulo giroscópio/acelerômetro MPU6050 que mede o ângulo dos eixos do satélite (*pitch*, *yaw* e *roll*) e o acelerômetro, que mede as forças aplicadas aos eixos do satélite; Módulo de umidade e temperatura DHT11, responsáveis por monitorar a umidade e temperatura do ambiente dentro do satélite verificando a necessidade de aquecer ou resfriar o ambiente interno (os testes de resistência do satélite irão comprovar se o

isolamento térmico, elétrico e magnético garantem a manutenção da temperatura interna, caso negativo, iremos colocar uma placa de peltier para o controle de temperatura). Os dados são enviados ao processador e armazenados usando um identificador alfabético para o envio a estação de telemetria..

3. - O experimento envolve o uso de uma câmera térmica e a memória SD Card de alta capacidade. A memória é utilizada para o armazenamento dos dados de telemetria e da câmera térmica, com o processador interpretando as imagens e armazenando apenas a posição geográfica dos pontos com óleo encontrados. A câmera térmica deve monitorar a temperatura da água e variações de 0,2 °C vão identificar possível presença de óleo sobre água. Ainda está sendo discutido sobre os tipos e resoluções das câmeras comerciais para determinar a que melhor e mais acessível ao projeto.
4. - O microcontrolador ESP8266 é responsável pela coleta dos dados de telemetria e experimento, salvando os dados na memória. O ESP8266 contém um RTC(Real time clock) que informará a data e horário da imagem da suposta mancha de óleo facilitando na obtenção dos dados em campo. Além disso, o processador será responsável pela interpretação da imagem, e assim, reduzimos o tamanho do pacote que será enviado pelo sistema de transmissão.
5. - A área de telecomunicações é responsável pela comunicação entre o satélite e o receptor na terra. Todos os dados salvos na memória flash serão enviados pelo módulo *LORAWAN* para o receptor em campo. Estes dados serão convertidos da linguagem C# do ESP em uma pacote de dados JSON, e deve conter um código de 4 letras e 4 números que atuam como identificador do satélite.

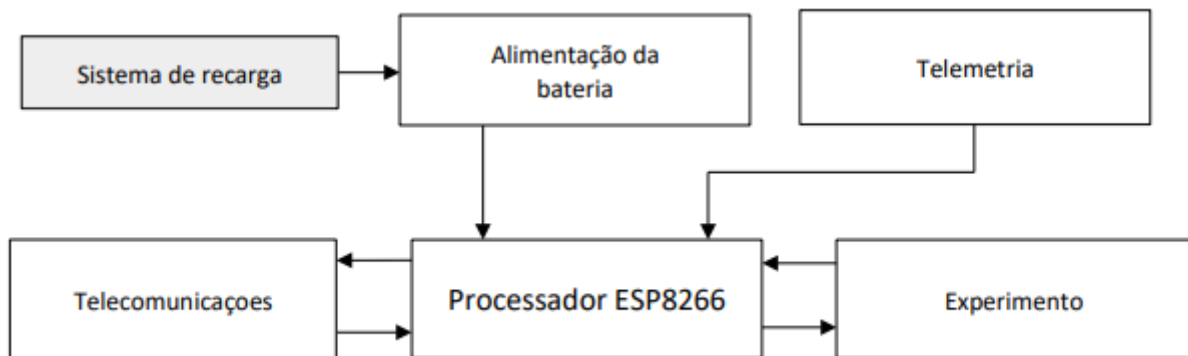


Figura 3 – Fluxograma dos sistemas do satélite
Fonte: autoria própria

5.2. Procedimento de execução da missão

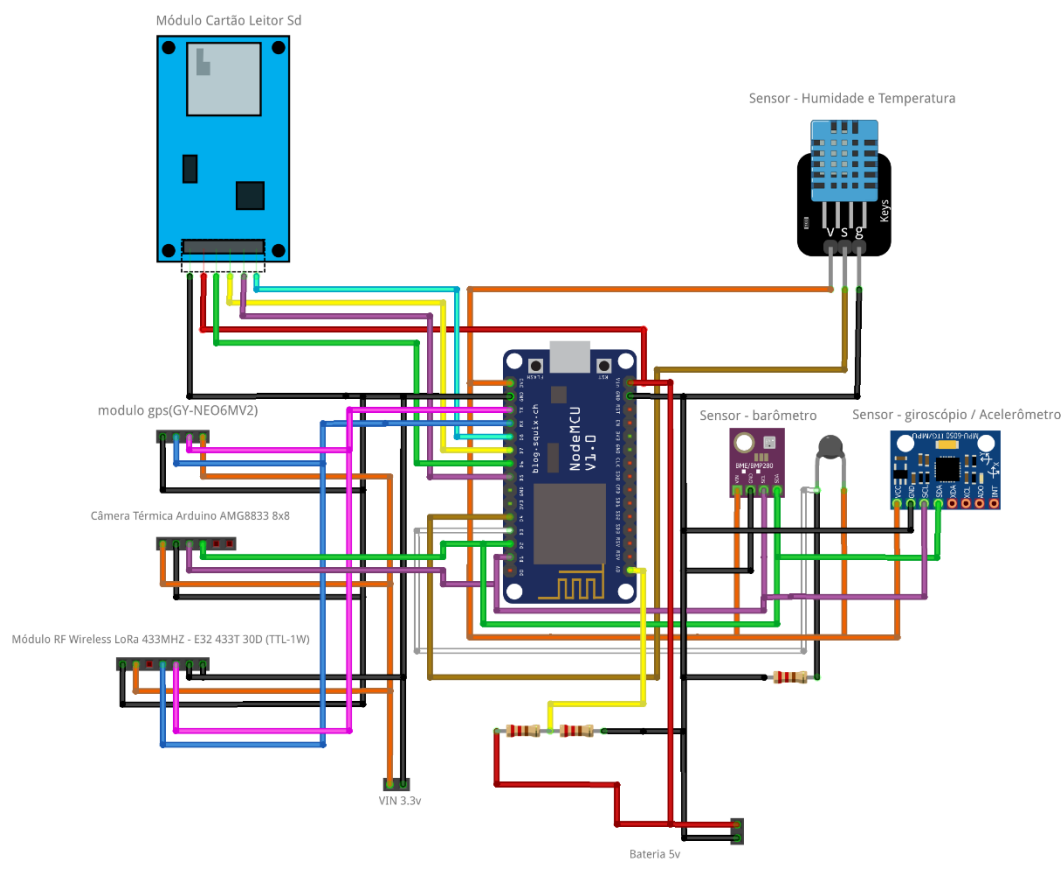


Figura 4 – Esquema elétrico do satélite
Fonte: autoria própria

A placa do satélite será montada artesanalmente utilizando uma placa universal. Uma placa não tem espaço suficiente para colocar todos os componentes necessários para o funcionamento do satélite, então pensamos em fazer um “sanduiche” de placas colocando as placas uma sobre a outra conectados por barras de pinos, e fixadas na estrutura com apenas quatro parafusos. Este sistema será conectado a estrutura do satélite e espumas serão usadas para amortecer possíveis impactos e evitar que os componentes se soltem após um choque mecânico.



Figura 5 – Modelo da placa de telemetria do satellite
Fonte: autoria propria

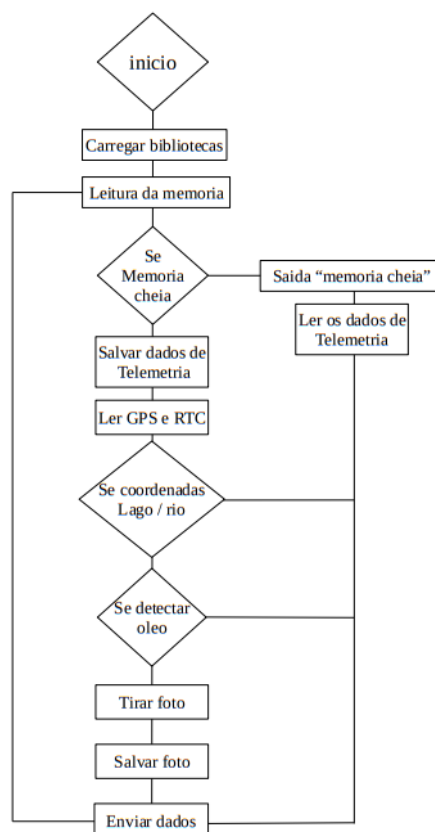


Figura 6 – fluxograma de programação do satellite
Fonte: Autoria propria

O satellite será programado o em C++ na plataforma IDE do Arduino (Anexo A) usando as respectivas bibliotecas e depois fazendo a integração com o BIPES. Após carregar as bibliotecas, uma checagem de memória será enviada pelo satellite e, se a memória estiver cheia, somente serão enviados os dados da telemetria com o status do satélite - posição e um aviso de memória cheia. Se tiver espaço para salvar, ele gravara na memória os dados da telemetria endereçado usando letras para identificar os sensores e números para os valores. Pontos específicos da rota orbital do satélite serão gravados onde são identificados áreas de lagos ou rios e, quando o GPS identificar estes pontos, serao tiradas fotos e o processador vai interpretar as cores e verificar se tem manchas de óleo, gravando a posição em que as

manchas foram encontradas. Quando o satélite passar pela estação de monitoramento em terra, esses dados serão enviados com as possíveis localizações destas manchas.

Os dados captados pelo sensor serão processados por um algoritmo denominado KNN (do inglês, *K-Nearest Neighbor*), que utiliza referências de cores para detectar possíveis manchas de óleo. O algoritmo funcionará da seguinte forma: um ponto de referência é escolhido e comparado com os pontos adjacentes, sendo que a diferença de cor indica a possível presença de uma mancha de óleo.

Para a transmissão de dados da imagem, houve a necessidade da divisão da imagem para ela ser transmitida. Com isso, variáveis foram criadas para o computador saber que é uma imagem que será dividida (é nele que a imagem vinda do satélite irá ser armazenada em base64) e a divisão para cada parte da imagem.

A função `dividir4` foi criada para dividir e criar links na base64 para cada divisão, utilizando a incrementação, que foi criada "a partir do ponto 0 da imagem" até a quarta parte dividida, do comando `for(var i=0; i<4; i++)`.

Após isso, todas as partes da imagem são separadas em colunas numa variável em JSON que são exibidas em `console.log` no seguinte jeito: todas as partes, `coluna1`, `coluna2` e cada pedaço da imagem. (Anexo B)

6. Materiais e métodos

Tabela 3- Comparativo entre memória flash e cartão SD

Característica	Memória W25Q64	Módulo Cartão + Cartão SD
Capacidade	64 Mb (157 imagens)	64Gb (157.000 imagens)
Tensão de operação	2,7 a 3,3 V	3,3 a 5 V
Corrente	4 mA	6 mA
Peso	1,6 g	7g
Custo	R\$15,17	R\$29,70
Número de pinos utilizados	4	4
Interface	SPI	SPI

Tabela 4 – Comparativo entre câmeras/sensor infravermelho

Característica	AMG 68833	MLX 90640	MLX 90614
Tensão de operação	3-5V	3-5V	3-5V
Corrente	120 mA	25 mA	5 mA
Faixa de medição	0 a 80 °C	-40 a 380 °C	-70 a 360 °C
Erro	± 2,5 °C	± 1 °C	± 0,5 °C
Resolução	0,2 °C	0,15 °C	0,02 °C
Resolução imagem	8x8 pixels	32x24 pixels	sensor
Campo de visão	55°	55x35°	90 ou 35 ou 10°
Peso	2 g	4 g	1,4 g
Custo	R\$ 232,86	R\$ 385,24	R\$ 129,90
Interface	I2C	I2C	I2C/TWI

Tabela 5 – Comparativo entre dispositivos *LORA*

Modelo	Potência	Frequência	Sensibilidade	Distância horizontal atingida	Taxa de comunicação	Peso (com antena)
SX1278	1W	433 Mhz	-148 dBm	8 km	37500 BPS	13 g
SX1276	0,1 W	433 Mhz	-130 dBm	3 km	300 000 BPS	12,2 g
RN2903	0,3W	915 Mhz	-146 dBm	15 km	12500 BPS	12,4 g
RFM95W	0,5 W	915 Mhz	-139 dBm	8 km	37500 BPS	15 g

Tabela 6 – Comparativo entre baterias para alimentação do satélite

Modelo	Tensão	Corrente	Dimensões	Peso
Li-Ion 18650	3,7 V	2,2 A	65x18 mm	45g
Lipo	7,4 V	2,2 A	18x32x100 mm	117 g
NiCD*	1,2 V	2,5 A	25x50 mm	11 g
NiMh	7,2 V	2,7 A	52x54 mm	139 g

* Não recarregável

Tabela 7 – Comparativo entre PLA e PETG

Fonte: (SANTANA et al, 2018).

PROPRIEDADES	PLA	PETG
$\sigma_{\text{máx}}$ (MPa)	$53,32 \pm 0,33$	$49,78 \pm 0,37$
$\epsilon_{\sigma\text{máx}}$ (mm/mm)	$0,02 \pm 2,6\text{E-}4$	$0,05 \pm 2,0\text{E-}4$
E (GPa)	$2,69 \pm 0,03$	$1,50 \pm 0,02$
ν (u.a)	$0,30 \pm 0,08$	$0,43 \pm 0,001$
$Massa$ (g)	$10,40 \pm 0,03$	$10,78 \pm 0,02$

A estrutura do satélite será construída com material PLA, contendo placas internas compostas por camadas de materiais: papel *Contact* de aeromodelismo, papelão, papel alumínio e isopor e uma outra de *Contact*. Essas camadas são responsáveis pelo isolamento térmico, elétrico e pela minimização de interferências eletromagnéticas, além de aumentar a resistência do satélite. A camada externa será de alumínio aeronáutico, seguindo as normas da competição.

Para os testes do satélite, serão utilizadas referências de equipes que já participaram da OBSAT e serão adaptadas às necessidades específicas deste projeto.

Tabela 8 – Média do preço dos materiais em 2023

Materiais	Custo
Esp 8266	23,50
Giroscópio MPU6050	19,86
Barômetro BMP180	13,59
GPS NEO6MV2	66,41
DHT11	15,01
NTC	68,78
Módulo Leitor de Cratão SD micro	8,88
Cartão microSD Card 64G	29,90
Câmera IR MLX90640	74,95
<i>LORAWAN</i> + antena*	129,90

*ainda estamos estudando o uso de um amplificador de sinal na antena para melhorar o sinal

7. Requisitos e restrições do projeto

Tabela 9- Requisitos globais da missão

Fonte: Autoria propria

Item	Requisito
Dimensões externas	1U 10x10x10cm
Condições de operação	Temperatura -80°C Vibração 230Hz por 1 minuto
Massa total	450g
Potência total	3,7 W
Sistema de telemetria	Acelerômetro 3 eixos ± 2 , ± 4 , ± 8 , $\pm 16g$ Giroscópio 3 eixos ± 250 , 500, 1000, 2000°/s Barômetro 300hPa a 1100hPa resolução 50cm
Tamanho do armazenamento a bordo	Acima de 4Gb
Altitude de operação	30km
Campo de visão da câmera	55° a 125°
Resolução temporal	10 FPS a 30 FPS
Comprimento de onda desejável	Infravermelho Próximo de 700 a 2500nm
Sensibilidade da câmera	640x380px, Resolução 0,2°C
Material estrutural	PLA ou PETG
Comunicação	40min intervalo de 4 em 4 minutos Limite de 90bytes Frequências: 433Mhz ou 915Mhz

Um dos desafios encontrados pela equipe foi a escolha entre sensores e câmeras térmicas. Embora os sensores parecessem uma opção viável, eles não garantiam a identificação correta do óleo, tornando a câmera térmica a escolha mais acertada. No entanto, a questão da distância apresentou-se como um obstáculo para a efetividade do projeto.

Para solucionar este problema, foi necessário calcular a resolução da câmera por área visualizada, chegando a uma resolução de 640x320 com um zoom de 100x, resultando a um valor aproximado de 38 m2 por pixel, com satélite a 30 km de altitude.

Outro desafio será ajustar um conjunto de lentes a câmera e ao tamanho reduzido do satélite, de forma que consigamos ajustar o foco a altitude do terreno que pode variar.

Outro desafio enfrentado foi o gerenciamento de energia, já que o projeto dependia de baterias para seu funcionamento. Será necessário calcular a vida útil da bateria e dimensionar, segundo o peso da carga útil do satélite, para a maior duração da missão.

Por fim, o custo do projeto foi um problema significativo, uma vez que o valor de cada peça necessária para o satélite é elevado. Os orçamentos estão em andamento para encontrar soluções viáveis para a realização do projeto que se adequem às especificações da missão proposta. Além disso estamos buscando formas de financiamento para construção. Hoje temos 60% do material e estamos trabalhando numa solução para câmera térmica.

Em relação a programação, encontramos farto material em inglês e o processo de adequação desse material as nossas necessidades têm se mostrado um desafio relativamente fácil, graças a experiência de nossa equipe em competições de robótica (Olimpíada Brasileira de Robótica e Mostra Nacional de

Robótica).

8. Cronograma preliminar de desenvolvimento e plano de trabalho.

Tabela 10 – Cronograma geral da fase 1 do projeto

Semana	Atividades: Fase 1 do projeto	Responsável/Coordenador
06 a 13/02	Montagem da Equipe Definição do nome da equipe Objetivo do satélite	Alan e Daniel
13 a 17/02	Busca de bibliografia do tema Preparação do plano B Discussão das Ideias – Brainstorming Divisão das áreas de pesquisa	Daniel Participantes: Pedro, Gabriel 1, Ana Laura e Alan
20 a 24/02	Retomada das atividades do projeto Discussão sobre a bibliografia: apresentação de alternativas Retomada das regras de montagem	Ana Daniel, Gabriel 1 e Pedro
27/02 a 03/03	Preparação da descrição do experimento Apresentação dos cálculos de gastos/investimento Descrição do teste de resistência térmica e mecânica (estresse) Elaboração do esquema eletrônico – Desenho da placa	Ana Gabriel 1 Pedro Daniel
06 a 10/03	Revisão do artigo: Introdução e Metodologia Apresentação do cálculo de gasto de energia e opções de alimentação (como recarregar?) Testes de estresse: desenho do modelo Programação do sistema de telemetria: sensores	Ana Gabriel 1 Pedro Daniel
13 a 17/03	Artigo: Metodologia Sistema de transmissão de dados: RF Testes de estresse: Desenho do modelo	Ana Gabriel 1 Pedro
20 a 24/03	Artigo: Análise de dados/Cronograma Fase 2 e Fase 3 Sistemas de Transmissão de dados: RF Testes de estresse: Impressão do modelo Ver 1	Ana Gabriel 1 Pedro
27 a 31/03	Finalização do projeto para apresentação: Formatação ABNT Teste de envio de pacotes json – incorporação no sistemas de controle Revisão do modelo ver 1/ Testes de estresse	Ana/Daniel Gabriel 1/Daniel Pedro
03/04 a 07/04	Montagem da apresentação	TODOS
Final da Fase 1		

Tabela 11 – Cronograma da fase 2 Para Eletronica

Divisão		Telemetria					Bateria		Experimento		Comunicação
Eletrônica e robótica	Etapas	MPU6050	BMP280	DHT11	RTC	GPS	NTC	Divisor de tensão	Memória W25q64	Sensor/Câmera Térmica	Lorawan
Daniel	Testes de bancada	Março	Março	Abril	Abril	Abril	Maior	Maio	Abril	Junho	Junho
	Montagem do	Março	Março	Abril	Abril	Abril	Maior	Maio	Abril	Junho	Junho

	Circuito										
	Montagem da placa	Março	Março	Abril	Abril	Abril	Mai o	Maio	Abril	Junho	Junho
	Testes de campo	Março	Março	Abril	Abril	Abril	Mai o	Maio	Abril	Junho	Junho

Tabela 12 – Cronograma da fase 2 Para Engenharia

Engenharia Pedro	Etapas	Março	Abril	Maio	Junho	Julho	Agosto	Setembro	Outubro
	Estudo das regras/Revisão de literatura								
	Pesquisa da estrutura								
	Impressão em PLA								
	Metodologia dos testes								
	Testes sem eletrônica								
	Testes com eletrônica								
	Homologação Correções de projeto								
	Plano de Extensão da missão								

Tabela 13 – Cronograma da fase 2 Para Experimento

Experimento Ana Laura	Etapas	Março	Abril	Maio	Junho	Julho	Agosto	Setembro	Outubro
	Revisão da literatura								
	Montagem do projeto								
	Metodologia do projeto								
	Definição das câmeras								
	Estudo Matemático de resolução								
	Análise de dados preliminar (testes de bancada)								
	Conclusão								

Tabela 14 – Cronograma da fase 2 Para Telecomunicações

Telecomunicações Gabriel	Etapas	Março	Abril	Maio	Junho	Julho	Agosto	Setembro	Outubro
	Pesquisa literatura								
	Estudo do BIPES								
	Programa preliminar								
	Envio de imagens Interpretação de imagens Tamanho do pacote								
	Segurança dos dados								
	Integração dos Sistemas								
	Testes de campo								

Anexos

Anexo A – Código de telemetria TSD 3.9 (telemetria de salvamento de dados)

```
//delay(1000) é um segundo
//essa versao do codigo esta com led ; True
//tempo antes da leitura = 1 min
//espacamento de leitura = 10ms
//tempo de leitura = 4 segundos
//numero maximo de leituras 1L
//-----//
// bibliotecas //
#include <FS.h>
#include <MPU6050_tockn.h>
#include <Wire.h>
//#include "MS5611.h"
//#include "SPIFFS.h"
#include <Adafruit_Sensor.h> //INCLUSÃO DE BIBLIOTECA
#include <Adafruit_BMP280.h> //INCLUSÃO DE BIBLIOTECA

//=====//

// variaveis de fisico //
long ler = 0; // essa variavel define quando o esp vai ler
int bota = 02; // porta de entrada do botão [ler]
int led = 13; // liga / desliga led
int desl = 0; //liga / desliga o modo somente leitura
int maxmetros;
int botas = 14;

//-----//
double vertical;
```

```

double horizontal;
double eixoZ;
//-----//
int voo = 0; // coordenada [y]
String ant; // string valor anterior
int maximoVoo = 1;
int comp;
//-----//

// sensores //
MPU6050 mpu6050(Wire);
Adafruit_BMP280 bmp; //OBJETO DO TIPO Adafruit_BMP280 (I2C)

uint32_t start;
uint32_t stop;
//-----//

//=====gravando e lendo=====//

//gravar arquivos no esp -----
void gravar(String state, String path)
{
    File rFile = SPIFFS.open(path,"w+");
    if(!rFile)
    {
        Serial.println("erro 01 - arquivo nao encontrado");
    }
    else
    {
        rFile.println(state);
    }
    rFile.close();
}

// ler os arquivos esp
String leroarq(String path) // ler o arquivo x
{
    File rFile = SPIFFS.open(path,"r");
    if(!rFile)
    {
        Serial.println("erro 01 - arquivo nao encontrado");
    }
    String content = rFile.readStringUntil('\r');
    //Serial.println(content);
    rFile.close();
    return content;
}

//abrir o spiffs -----
void openFS(void){
    //Abre o sistema de arquivos
    if(!SPIFFS.begin()){
        Serial.println("\nErro ao abrir o sistema de arquivos");
    } else {
        Serial.println("\nSistema de arquivos aberto com sucesso!");
    }
}

```



```
//=====setup=====//

void setup() {

  Serial.begin(115200);

  Serial.println("wire begin");
  Wire.begin();

  Serial.println("MPU6050 begin");
  mpu6050.begin();

  Serial.println("bmp begin");
  bmp.begin(0x76);

  Serial.println("mpu calculando");
  mpu6050.calcGyroOffsets(true);
  pinMode(bota,INPUT);
  pinMode(led,OUTPUT);
  openFS();
  digitalWrite(led,HIGH);
  pinMode(botas,OUTPUT);
  digitalWrite(botas,HIGH);

  Serial.println("-----");
  Serial.println("ola! :)");
  Serial.println("comandos: ler - (ler os arquivos) | fd - (formatar disco) | rl - (relatorio dos voos) | ? - (exibir
comandos) | voo - (numero de voos) | sml - (somente leitura liga/desliga)");
  ant = leroarq("/leitura.txt");
  voo = leroarq("/voos.txt").toInt();
  Serial.println("voos salvos: " + String(voo));
  Serial.println ("somente leitura: "+String(desl));
  ledL();
}

//=====loop=====//

void loop()
{
  //-----comandos-----//

  if(Serial.available())
  {
    String command = Serial.readStringUntil('\n');

    if (command == "ler" && desl == 0) // leitura -----
    {
      if(ler == 0 && voo < maximovoo)
      {
        ler = 1;
        Serial.println("ler = 1");
      }
      else if(ler == 1)
      {
        Serial.println("ainda lendo...");
      }
      else

```

```

    {
        led01();
    }
}

if (command == "fd")// formatar -----
{
    ant = "";
    voo = 0;
    SPIFFS.remove("/voos.txt");
    SPIFFS.remove("/leitura.txt");
    Serial.println("disco formatado com sucesso!");
}

if (command == "rl")// relatorio -----
{
    String leitura = leroarq("/leitura.txt");
    Serial.println(leitura);
}

if (command == "sml")// somente leitura -----
{
    if(desl == 0)
    {
        desl = 1;
        delay(10);
    }
    else
    {
        desl = 0;
        delay(10);
    }
}

if (command == "?")// nao sei os comandos -----
{
    Serial.println("comandos: ler - (ler os arquivos) | fd - (formatar disco) | rl - (relatorio dos voos) | ? - (exibir comandos) | voo - (numero de voos) | sml - (somente leitura liga/desliga)");
}

if (command == "voo")// voos -----
{
    Serial.println("voos salvos: " + String(voo));
}
}

//-----botão-----//

int lig = digitalRead(bota); // variavel estado do botão (NÃO MEXER)
if(lig == 1 && desl == 0)
{
    if(ler == 0 && voo < maximovoo)
    {
        ler = 1;
        Serial.println("ler = 1");
    }
    else if(ler == 1)
    {

```

```

        Serial.println("ainda lendo...");
    }
    else
    {
        led01();
    }
}

//-----gravando os arquivos-----//
if (desl == 0)
{
    if(ler == 1)//O INT LER É PARA COMEÇAR A LEITURA
    {
        ant=( ant + '\n' + "-----" + "VOO " + String(voo + 1) + "-----" ); //mostrar valor do voo

        ant = (ant + '\n' + "barometro inicio: " + String(bmp.readAltitude(1013.25),0));
        ant = (ant + '\n' + "bar" + '\t' + "Ax" + '\t' + "Ay" + '\t' + "Az" + '\t' + "Gx" + '\t' + "Gy" + '\t' + "Gz" );

        for(int i = 0; i <= 150 ; i++)// i <= 150 //
        {
            ledH();
            delay(200);
            ledL();
            delay(200);
        }

        ledH();
        for(int i = 0; i <= 100 ; i++)
        {

            girinho();
            String sensores = ( String(bmp.readAltitude(1013.25),0) + '\t' + String(mpu6050.getAccX()) + '\t' +
String(mpu6050.getAccY())+ '\t' + String(mpu6050.getAccZ()) + '\t' + String(horizontal) + '\t' +
String(vertical) + '\t' + String(eixoZ));
            String numero = (String(i) + '\t' );

            ant =(ant + '\n' + sensores);

            gravar(ant, "/leitura.txt");
            delay(100);
        }

        ledL();
        ant = (ant + '\n' + "-----");
        voo ++;
        gravar(String(voo),"/voos.txt");
        ler = 0;
        Serial.println("l.p");
    }
}

else //--- SOMENTE LEITURA
{
    girinho();
    //Serial.print(bmp.readAltitude(1013.25),0);
    Serial.println(String(mpu6050.getAngleX()) + '\t' + String(mpu6050.getAngleY()) + '\t' +
String(mpu6050.getAngleZ()));
}

```

```

}
}
//----- sinais -----//

void ledL() // led desliga
{
  digitalWrite(led,LOW);
}

void ledH() // led liga
{
  digitalWrite(led,HIGH);
}
//----- erros -----//

void led01() // erro 01 - memoria maxima exedida
{
  Serial.println("memoria maxima exedida...");
  for(int i = 0; i <= 2 ; i++)
  {
    delay(200);
    digitalWrite(led,LOW);
    delay(200);
    digitalWrite(led,HIGH);
  }
}

void girinho()
{
  mpu6050.update();
  horizontal = (atan2(mpu6050.getAngleX(),mpu6050.getAngleZ()) + PI) * RAD_TO_DEG;
  vertical = (atan2(mpu6050.getAngleX(),mpu6050.getAngleY()) + PI) * RAD_TO_DEG;
  eixoZ = (atan2(mpu6050.getAngleY(),mpu6050.getAngleZ()) + PI) * RAD_TO_DEG;
  //delay (500);
}

```

Anexo B – Código de transmissão de dados V1.0

```

// Criação do canvas, variável onde ficará salva as partes divididas da imagem e a imagemem si
var canvas = document.createElement('canvas'),
    ctx = canvas.getContext('2d'),
    parts = [],
    img = new Image();

img.onload = dividir4;

function dividir4() {
  for(var i=0; i<4; i++) {
    // Conversão do canvas para base64
    parts.push(canvas.toDataURL());
  }

  // Cria uma variável paraa armazenar o JSON, com a base64 de cada elemento repartido da
  imagem

```

```

var JSON = {
  'coluna1': parts[0] + '\n' + parts[2],
  'coluna2': parts[1] + '\n' + parts[3]
}

// Exibição do JSON no console
console.log(JSON)
console.log(JSON['coluna1'])
console.log(JSON['coluna2'])
console.log(parts[0])
console.log(parts[1])
console.log(parts[2])
console.log(parts[3])

}

// Imagem que será dividida
img.src = "" //texto grande demais para o artigo

```

Referências

Ecóleo, “**Associação brasileira para sensibilização, coleta e reciclagem de resíduos de óleo comestível**”, 2023, <https://ecoleo.org.br/> Acesso em 09 de abril de 2023.

OilWorld, “**Independent Global Market Analyses & Forecasts**”, 2023, <https://www.oilworld.biz/> Acesso em 09 de abril de 2023.

Sabesp, “**Companhia de Saneamento Básico do Estado de São Paulo**”, 2023 , <https://www.sabesp.com.br/site/Default.aspx> Acesso em 09 de abril de 2023.

MORAES, Paula Louredo. 2022, "O que é e quando ocorre a mutação"; *Brasil Escola*. Disponível em: <https://brasilestela.uol.com.br/biologia/mutacao.htm>. Acesso em 09 de abril de 2023.

Recicla Sampa, “**Recicla Sampa**”, 2023 , <https://www.reciclasampa.com.br/> Acesso em 09 de abril de 2023

JÚNIOR, Mário Roberto Lemos, **A indústria de petróleo e os derramamentos de óleo no mar: uma abordagem exploratória, universidade federal fluminense escola de engenharia departamento de engenharia química e de petróleo curso de engenharia de petróleo**, Niterói, 2013

SANTANA, Leonardo et al. Estudo comparativo entre PETG e PLA para Impressão 3D através de caracterização térmica, química e mecânica, 2018, Disponível em: <https://www.scielo.br/j/rmat/a/dpWDvBJzSXYtzbKnJdDqHVg/?lang=pt&format=html#> Acesso em 09 de abril de 2023.

Inspired Engineering, “**MLX90614 family Single and Dual Zone Infra Red Thermometer in TO-39.**”, Melexis, 2019, <https://www.melexis.com/en/documents/documentation/datasheets/datasheet-mlx90614> Acesso em 09 de abril de 2023

Fabio Guimarães, Voltímetro, Amperímetro e Ohmímetro c/ Arduino, mundoprojetado, 2018 Disponível em: <https://mundoprojetado.com.br/voltmetro-amperimetro-e-ohmimetro-c-arduino/> Acesso em 09 de

abril de 2023

José Moraes, ESP32 – Utilizando o RTC interno do ESP32, Vida de Silício, 2019 Disponível em: <https://portal.vidadesilicio.com.br/esp32-utilizando-o-rtc-interno-para-datas/> Acesso em 09 de abril de 2023

INPE, Câmeras Imageadoras CBERS-3 e 4, CBERS, 2019 Disponível em: <http://www.cbers.inpe.br/sobre/cameras/cbers3-4.php> Acesso em 09 de abril de 2023

Alex Newton, DIY Thermal Camera with ESP8266 & AMG8833 Thermal Image Sensor, Integrating AMG8833 8X8 Array Thermal Camera with ESP8266, how2eletronics, 2023 Disponível em: <https://how2electronics.com/diy-amg8833-thermal-camera-with-esp8266-ili9341/> Acesso em 09 de abril de 2023

Fernando Koyanagi, Infrared Sensor With ESP8266, AUTODESK Instructables, 2018 Disponível em: <https://www.instructables.com/Infrared-Sensor-With-ESP8266/> Acesso em 09 de abril de 2023

Floris Wouterlood, 8*8 pixel AMG8833, thermal sensor, ESP8266 microcontroller board and a SSD1283 driven TFT display, The Solar Universe, 2022, Disponível em: <https://thesolaruniverse.wordpress.com/2022/05/02/88-pixel-amg8833-thermal-sensor-esp8266-microcontroller-board-and-a-ssd1283-driven-tft-display/> Acesso em 09 de abril de 2023

Djames Suhanko, Transferir uma imagem jpg por serial – ESP8266 e ESP32, dobitaobyte 2019 Disponível em: <https://www.dobitaobyte.com.br/transferir-uma-imagem-jpg-por-serial-esp8266-e-esp32/> Acesso em 09 de abril de 2023

PAPER, Conference, Desenvolvimento de um cubesat para detecção de descargas atmosféricas: projeto raiosat, 2019

RABELO, Leila Baganha, Estudo da variabilidade da estrutura vertical da temperatura da água e da profundidade da termoclina na região da confluência brasil-malvinas, 2010

Inspired Engineering, “*MLX90614 family Single and Dual Zone Infra Red Thermometer in TO-39.*”, Melexis, 2019, [Datasheet for MLX90614 I Melexis](#) Acesso em 09 de abril de 2023

TEMBA, Plínio, Fundamentos da Fotogrametria, Departamento de Cartografia, UFMG, Geoprocessamento, 2000

Natanael José Maciel Isidoro (Estudante), Ludmila de Oliveira Agra (Estudante), Pedro Hugo Ursulino Fernandes (Estudante), Rodrigo Queiros de Almeida (Orientador), A²Database (Araripe Atmospheric Database): Plataforma de Monitoramento de Dióxido de Carbono e Compostos Orgânicos Voláteis na Chapada do Araripe, IFCE Campus Juazeiro do Norte, 2022

JÚNIOR, Mário Roberto Lemos, A indústria de petróleo e os derramamentos de óleo no mar: uma abordagem exploratória, universidade federal fluminense escola de engenharia departamento de engenharia química e de petróleo curso de engenharia de petróleo, Niterói, 2013