



MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÕES
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

**GERENCIAMENTO DE BASE DE DADOS RADIOMÉTRICOS E DE
QUALIDADE DE ÁGUA EM UM BANCO DE DADOS POSTGIS NO ÂMBITO
DO LABISA-INPE**

Guilherme Yokota Sato

Relatório de Iniciação Científica do
programa PIBIC orientado por Cláudio
Clemente Faria Barbosa.

URL do documento original:
<<http://urlib.net/>>

INPE
São José dos Campos
2025

RESUMO

Durante o ano de 2025, duas frentes de desenvolvimento foram conduzidas referente às atividades do Laboratório de Instrumentação de Sistema Aquáticos (LabISA). A primeira consistiu na atualização do site institucional do laboratório, com foco na ampliação e organização do conteúdo científico e institucional. Foram incluídos projetos em andamento e a divulgação de publicações recentes dos pesquisadores, tornando o site uma ferramenta mais eficaz para a comunicação com a comunidade acadêmica e o público interessado. Essa iniciativa buscou valorizar a produção científica do laboratório e facilitar o acesso às informações sobre suas pesquisas. A segunda frente, teve como objetivo a reorganização de rotinas utilizadas no processamento de dados brutos, como a simulação de bandas espectrais, o cálculo de CDOM (matéria orgânica dissolvida colorida) e a estimativa de ficocianina. Essas rotinas foram refatoradas para serem utilizadas como funções reutilizáveis, com entradas e saídas padronizadas, visando facilitar sua integração ao *backend* da aplicação. Essa refatoração permitiu maior modularidade e reutilização do código, alinhando-se às boas práticas de desenvolvimento de software científico. Além disso, foi realizada a integração entre o *frontend* e os serviços responsáveis por acionar essas funções, permitindo a execução automatizada das rotinas a partir de uma interface web. Essa funcionalidade compõe a base da aplicação LIA (*LabISA Interactive Assistant*), um assistente interativo acessado via navegador, que possibilita a execução das rotinas por pesquisadores, estudantes e colaboradores. A interface do LIA foi desenvolvida com foco em simplicidade, usabilidade e acessibilidade, promovendo maior eficiência nos fluxos de trabalho do laboratório e ampliando o alcance das ferramentas.

Palavras-chave: Rotinas automatizadas. Sensoriamento remoto. Refatoração de código. Interface Web. Comunicação científica.

LISTA DE FIGURAS

	<u>Pág.</u>
Figura 1 – Apresentação do código em R	11
Figura 2 - Apresentação do código em Python	12
Figura 3 - Apresentação do código de simulação de bandas	13
Figura 4 – Apresentação do código de CDOM	14
Figura 5 – Apresentação do código de estimativa de ficocianina	14
Figura 6 – Apresentação do código de mostrando a padronização	15
Figura 7 – Interface do usuário	16
Figura 8 – Código de integração entre frontend e backend	16

SUMÁRIO

	<u>Pág.</u>
1 INTRODUÇÃO	12
FUNDAMENTAÇÃO TEÓRICA	22.1
22.2 TECNOLOGIAS DE BANCO DE DADOS ESPACIAIS	10
2.3 DESENVOLVIMENTO DE SOFTWARE	10
3 411	
3.1 5	12
3.2 613	
3.3 CÁLCULO DE CDOM Erro! Indicador não definido.	14
3.4 ESTIMATIVA DE FICOCIANINA Erro! Indicador não definido.	14
3.5 PADRONIZAÇÃO DE ENTRADAS E SAÍDAS Erro! Indicador não definido.	15
4 815	
4.1 815	
4.2 916	
5 CONCLUSÃO	17

1 INTRODUÇÃO

O monitoramento de sistemas aquáticos por sensoriamento remoto tem se tornado uma ferramenta fundamental para a compreensão e gestão dos recursos hídricos. No contexto do Laboratório de Instrumentação de Sistemas Aquáticos (LabISA) do INPE, a crescente demanda por análises de dados radiométricos e de qualidade da água exigiu o desenvolvimento de soluções tecnológicas mais eficientes e acessíveis.

Historicamente, o processamento desses dados dependia de rotinas individuais, resultando em duplicação de código, dificuldades de manutenção e limitações na reutilização. Além disso, o acesso a essas ferramentas era restrito a usuários com conhecimento técnico específico, limitando o alcance das pesquisas desenvolvidas no laboratório.

O presente trabalho propõe uma solução integrada para esses desafios, focando em duas frentes principais: a refatoração e modularização das rotinas de processamento existentes, e o desenvolvimento de uma plataforma web interativa que democratize o acesso a essas ferramentas. A implementação de um banco de dados PostGIS garante o gerenciamento eficiente dos dados espaciais e temporais, enquanto a criação do LIA (LabISA Interactive Assistant) proporciona uma interface intuitiva para execução das rotinas.

Este relatório documenta o processo completo de desenvolvimento, desde a análise das necessidades do laboratório até a implementação e validação do sistema final, demonstrando como a integração de tecnologias modernas pode potencializar a pesquisa científica em sensoriamento remoto de sistemas aquáticos.

2 FUNDAMENTAÇÃO TEÓRICA

O sensoriamento remoto de sistemas aquáticos baseia-se na análise da interação da radiação eletromagnética com os constituintes opticamente ativos presentes na água. Os dados radiométricos, obtidos através de instrumentos especializados como o TriOS-RAMSES, capturam assinaturas espectrais em alta resolução que permitem a inferência de parâmetros de qualidade da água através da aplicação de algoritmos bio-ópticos específicos. Esta abordagem tem se consolidado como ferramenta fundamental para o monitoramento ambiental, oferecendo vantagens significativas em termos de cobertura espacial, frequência temporal e custo-efetividade comparado aos métodos tradicionais de coleta in situ.

2.1 DADOS RADIOMÉTRICOS E QUALIDADE DA ÁGUA

Os dados radiométricos constituem a base fundamental para o monitoramento de sistemas aquáticos por sensoriamento remoto, baseando-se na interação da radiação eletromagnética com os constituintes opticamente ativos presentes na água. Instrumentos especializados como o TriOS-RAMSES capturam assinaturas espectrais em alta resolução que permitem a inferência de parâmetros de qualidade da água através de algoritmos bio-ópticos específicos.

A simulação de bandas espectrais envolve a convolução de dados hiperespectrais com funções de resposta espectral relativa (RSR) de sensores orbitais como Landsat-8, Sentinel-2 e MODIS. Este processo permite validar algoritmos localmente antes da aplicação em dados orbitais, facilitando o escalonamento de estudos locais para análises regionais e a comparação entre diferentes sensores.

A matéria orgânica dissolvida colorida (CDOM) representa compostos orgânicos caracterizados por forte absorção no azul e ultravioleta. Sua quantificação indica aportes terrestres, decomposição orgânica ou atividade microbiológica. Os algoritmos baseiam-se na absorção em 440nm ou razões espectrais específicas.

2.2 TECNOLOGIAS DE BANCO DE DADOS ESPACIAIS

O PostGIS representa uma extensão espacial robusta do sistema de gerenciamento de banco de dados PostgreSQL, oferecendo funcionalidades avançadas para armazenamento, indexação, consulta e análise de dados geoespaciais. Esta tecnologia tornou-se padrão na comunidade de sistemas de informação geográfica (SIG) devido à sua conformidade com padrões abertos do Open Geospatial Consortium (OGC) e sua capacidade de lidar com grandes volumes de dados espaciais de forma eficiente.

No contexto específico de dados radiométricos e de qualidade da água, o PostGIS oferece capacidades essenciais que incluem o armazenamento eficiente de geometrias pontuais representando locais de coleta, com suporte nativo a diferentes sistemas de coordenadas através da biblioteca PROJ. A indexação espacial automatizada utiliza estruturas de dados otimizadas como R-trees para garantir consultas rápidas baseadas em critérios espaciais, permitindo operações como busca por proximidade, interseção e contenção em tempos de resposta adequados mesmo para grandes conjuntos de dados.

2.3 DESENVOLVIMENTO DE SOFTWARE

O desenvolvimento de software científico apresenta características específicas que diferem do desenvolvimento comercial tradicional. Aspectos como reprodutibilidade, rastreabilidade de resultados e modularidade assumem importância crítica, uma vez que o software serve como ferramenta de análise e meio de validação de metodologias científicas.

A refatoração de rotinas científicas envolve a reestruturação do código existente sem alterar sua funcionalidade externa, visando melhorar a legibilidade, manutenção e reutilização. Este processo é fundamental para a evolução sustentável de ferramentas científicas, permitindo que conhecimento acumulado seja preservado e aprimorado, permitindo operações como busca por proximidade, interseção e contenção em tempos de resposta adequados mesmo para grandes conjuntos de dados.

A modularização permite criar componentes reutilizáveis que encapsulam conhecimento científico específico. Funções bem projetadas facilitam a adaptação de algoritmos para diferentes contextos, maximizando o retorno do investimento em desenvolvimento e promovendo padronização de metodologias.

A implementação de boas práticas inclui controle de versão, testes automatizados, documentação detalhada e interfaces padronizadas. Estas práticas garantem que modificações no código não comprometam a validade dos resultados e facilitam a colaboração entre pesquisadores.

3 REFATORAÇÃO DAS ROTINAS DE PROCESSAMENTO

A refatoração das rotinas tinha por objetivo, modularizar as funções para serem reutilizáveis e migrar de R para Python.

Figura 1 - Apresentação do código em R.

```
msi_simulation = function(spectra, point_name) {  
  
  #Simulação e Rrs  
  
  #Simula??o Sentinel 2A e 2B  
  
  require(openxlsx)  
  require(dplyr)  
  require(tidyr)  
  
  #L? as fun??es de resposta - SENTINEL 2A  
  S2_A_B1 <- s2_srf[,c(1,2)] %>% filter(SR_WL >= 400 & SR_WL <= 900)  
  S2_A_B2 <- s2_srf[,c(1,3)] %>% filter(SR_WL >= 400 & SR_WL <= 900)  
  S2_A_B3 <- s2_srf[,c(1,4)] %>% filter(SR_WL >= 400 & SR_WL <= 900)  
  S2_A_B4 <- s2_srf[,c(1,5)] %>% filter(SR_WL >= 400 & SR_WL <= 900)  
  S2_A_B5 <- s2_srf[,c(1,6)] %>% filter(SR_WL >= 400 & SR_WL <= 900)  
  S2_A_B6 <- s2_srf[,c(1,7)] %>% filter(SR_WL >= 400 & SR_WL <= 900)  
  S2_A_B7 <- s2_srf[,c(1,8)] %>% filter(SR_WL >= 400 & SR_WL <= 900)  
  S2_A_B8 <- s2_srf[,c(1,9)] %>% filter(SR_WL >= 400 & SR_WL <= 900)  
  S2_A_B8a <- s2_srf[,c(1,10)] %>% filter(SR_WL >= 400 & SR_WL <= 900)  
}
```

Figura 2 - Apresentação do código em Python.

```
def msi(self, spectra, point_names):
    band_indices = list(range(1, 10))
    wave_centers = [440, 490, 560, 665, 705, 740, 783, 842, 865]

    s2a_result = self._simulate_bands_direct_optimized(
        spectra, self.srf_data['s2a'], band_indices, wave_centers,
        point_names, wavelength_range=(400, 900)
    )

    s2b_result = self._simulate_bands_direct_optimized(
        spectra, self.srf_data['s2b'], band_indices, wave_centers,
        point_names, wavelength_range=(400, 900)
    )

    return {'s2a': s2a_result, 's2b': s2b_result}
```

3.1 METODOLOGIA DE REFATORAÇÃO

O processo de refatoração das rotinas existentes seguiu uma metodologia estruturada, baseada em princípios de engenharia de software e adaptada às necessidades específicas do ambiente científico. A abordagem adotada incluiu:

- **Análise das Rotinas Existentes:** Cada rotina foi analisada individualmente para identificar funcionalidades core, dependências externas, pontos de entrada e saída, e potenciais oportunidades de modularização;
- **Definição de Interfaces Padronizadas:** Foi estabelecido um padrão comum para entradas e saídas das funções, incluindo estruturas de dados, formatos de arquivos e convenções de nomenclatura;
- **Documentação Técnica:** Toda função refatorada recebeu documentação detalhada, incluindo descrição de propósito, parâmetros de entrada, valores de retorno e exemplos de uso.

O código refatorado seguiu convenções PEP 8 para Python, garantindo consistência e legibilidade. Além disso, foi implementado um sistema de *logging* para facilitar a depuração e monitoramento do desempenho.

3.2 SIMULAÇÃO DE BANDAS ESPECTRAIS

A simulação de bandas espectrais constitui uma das rotinas mais críticas do sistema, permitindo a conversão de dados hiperespectrais para configurações multiespectrais específicas de diferentes sensores.

Melhorias Implementadas:

- Suporte a múltiplos métodos de simulação;
- Interface configurável para adição de novos sensores;
- Validação automática de entrada de dados;
- Otimização computacional através de vetorização Numpy;

Figura 3 - Apresentação do código de simulação de bandas.

```
def olci(self, spectra, point_names):
    band_indices = list(range(1, 20))
    wave_centers = [400, 412, 442, 490, 510, 560, 620, 665, 673, 681,
                   708, 753, 761, 764, 767, 778, 865, 885, 900]

    return self._simulate_bands_direct_optimized(
        spectra, self.srf_data['s3'], band_indices, wave_centers,
        point_names, wavelength_range=(400, 900)
    )

def msi(self, spectra, point_names):
    band_indices = list(range(1, 10))
    wave_centers = [440, 490, 560, 665, 705, 740, 783, 842, 865]

    s2a_result = self._simulate_bands_direct_optimized(
        spectra, self.srf_data['s2a'], band_indices, wave_centers,
        point_names, wavelength_range=(400, 900)
    )

    s2b_result = self._simulate_bands_direct_optimized(
        spectra, self.srf_data['s2b'], band_indices, wave_centers,
        point_names, wavelength_range=(400, 900)
    )

    return {'s2a': s2a_result, 's2b': s2b_result}
```

3.3 CÁLCULO DE CDOM

A determinação de CDOM (Colored Dissolved Organic Matter) foi refatorada para suportar múltiplos algoritmos e permitir comparação de resultados entre diferentes metodologias.

Figura 4 - Apresentação do código de CDOM

```
def main():
    print("CDOM analysis application")

    # Run analysis
    try:
        results = run_cdom_analysis()

        if results["status"] == "success":
            print("\n Analysis completed successfully")
            sys.exit(0)
        elif results["status"] == "warning":
            print(f"\n Analysis completed with warnings: {results.get('message')}")
            sys.exit(0)
        else:
            print(f"\n Analysis failed: {results.get('message', 'Unknown error')}")
            sys.exit(1)

    except KeyboardInterrupt:
        print("\n Analysis interrupted by user")
        sys.exit(1)
    except Exception as e:
        print(f"\n Unexpected error: {e}")
        sys.exit(1)
```

3.4 ESTIMATIVA DE FICOCIANINA

A estimativa de ficocianina foi completamente reestruturada para incorporar avanços recentes na literatura científica e suportar diferentes tipos de corpos d'água.

Figura 5 - Apresentação do código de estimativa de ficocianina

```
def main():
    # Create analyzer instance
    analyzer = FicocianinAnalyzer()

    # Input file
    input_path = os.path.join("input", "medidas_promissao_ficocianina.txt")

    # Output path
    output_path = os.path.join("output", "ficocianina_results.csv")

    # Load data
    analyzer.load_data(input_path)

    # Preprocess data
    analyzer.preprocess_data()

    # Analyze samples
    analyzer.analyze_samples([[350, 350, 350, 300, 350, 350]])
```

3.5 PADRONIZAÇÃO DE ENTRADAS E SAÍDAS

Um aspecto fundamental da refatoração foi o estabelecimento de padrões consistentes para interfaces de função, garantindo interoperabilidade e facilidade de uso.

Figura 6 – Apresentação do código de mostrando a padronização

```
def _clean_spectra_data(self, spectra):
    # Replace negative values with 0
    negative_mask = spectra < 0
    spectra[negative_mask] = 0.0

    # Handle NaN values
    nan_mask = spectra.isna()
    nan_count = nan_mask.sum().sum()

    if nan_count > 0:
        spectra = spectra.fillna(0.0)
        print(f"Filled {nan_count} NaN values with 0.0")

    return spectra
```

4 DESENVOLVIMENTO DO LIA (LABISA INTERACTIVE ASSISTANT)

Após esses passos, iniciou-se o desenvolvimento do LIA, uma ferramenta com intuito de centralizar todas as rotinas computacionais presentes no laboratório.

4.1 ARQUITETURA DO SISTEMA

O LIA foi desenvolvido com uma arquitetura que garante escalabilidade, manutenção e flexibilidade para futuras expansões.

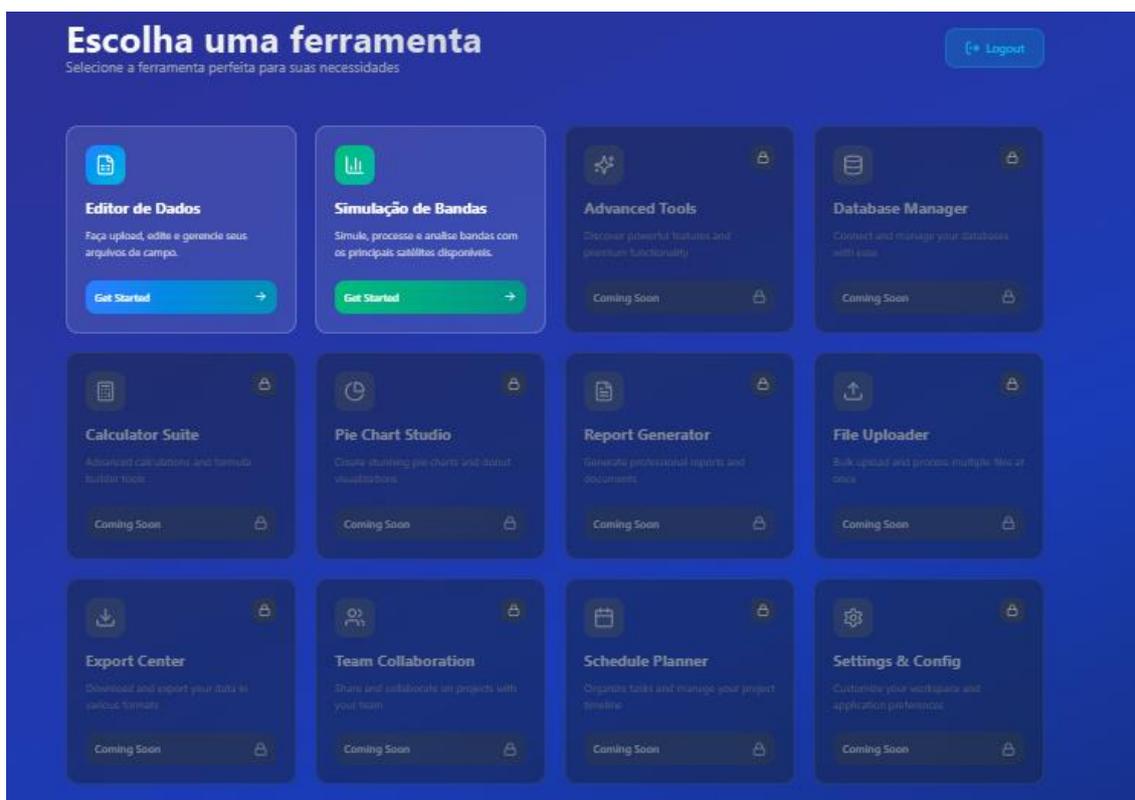
Componentes Principais:

- **Frontend (React.js):** Interface de usuário responsiva e intuitiva;
- **API Gateway (Django):** Gerenciamento de requisições e autenticação;
- **Serviços de Processamento (Python):** Execução das rotinas refatoradas;
- **Banco de Dados (PostGIS):** Armazenamento e consulta de dados.

4.2 INTERFACE DE USUÁRIO

A interface do LIA foi projetada com foco na experiência do usuário, considerando que os usuários finais incluem pesquisadores com diferentes níveis de familiaridade com a tecnologia.

Figura 7 – Interface do usuário.



4.3 INTEGRAÇÃO FRONTEND-BACKEND

A integração entre *frontend* e *backend* foi implementada seguindo padrões REST API e WebSocket para atualizações em tempo real.

Figura 8 – Código de integração entre frontend e backend.

```
// Make API call using axios
const response = await axios.post('http://localhost:8000/simulations/rotinaSimulacaoBandas/', formData, {
  headers: {
    'Content-Type': 'multipart/form-data',
  },
  timeout: 120000,
});

const result = response.data;
```

5 CONCLUSÃO

O desenvolvimento do sistema de gerenciamento de dados radiométricos e de qualidade da água no âmbito do LabISA-INPE atingiu todos os objetivos propostos, demonstrando melhorias significativas em eficiência, usabilidade e alcance das ferramentas científicas do laboratório.

Principais Contribuições:

- **Refatoração de Rotinas:** A transformação de rotinas ad hoc em funções modulares e reutilizáveis resultou em código mais eficiente de fácil manutenção e testes. A padronização de interfaces facilitou a integração entre diferentes componentes e reduziu significativamente a duplicação de código.
- **Plataforma LIA:** O desenvolvimento da interface web democratizou o acesso às ferramentas do laboratório, permitindo que usuários sem conhecimento técnico específico executem análises complexas.

Considerações Finais:

O projeto demonstra como a aplicação de práticas modernas de engenharia de software pode potencializar significativamente a pesquisa científica. A combinação de refatoração cuidadosa, tecnologias apropriadas e foco na experiência do usuário resultou em um sistema que não apenas resolve problemas técnicos imediatos, mas estabelece uma base sólida para o futuro desenvolvimento do laboratório.

A metodologia empregada pode servir como modelo para outros laboratórios enfrentando desafios similares, demonstrando que o investimento em infraestrutura tecnológica adequada gera retornos multiplicados em produtividade científica e qualidade de pesquisa.

O sucesso do projeto reforça a importância da colaboração interdisciplinar entre ciência da computação e ciências ambientais, evidenciando como

soluções tecnológicas bem projetadas podem acelerar descobertas científicas e ampliar o impacto da pesquisa em sensoriamento remoto de sistemas aquáticos.