

# Automatização do Processo de Geração de Mapas utilizando uma Arquitetura Orientada a Fluxos

Beatriz Barrios Caporusso<sup>1</sup>, Daniel dos Santos Kaster<sup>1</sup>

<sup>1</sup>Departamento de Computação – Universidade Estadual de Londrina (UEL)  
Caixa Postal 10.011 – CEP 86057-970 – Londrina – PR – Brasil

beatriz.caporusso@uel.br, dskaster@uel.br

**Abstract.** *Currently, data is considered an extremely valuable resource across various fields. Companies, organizations, and governments use this resource for decision-making and strategic planning. In agriculture, data can be used to generate maps, for example. The use of maps is very important for agriculture as they provide valuable information about various factors that directly impact agricultural production. Therefore, this project proposes to automate the generation of maps, allowing users to benefit from an automatic, fast, and accurate process, replacing the current manual, error-prone, and slower work. The proposal includes the implementation of a flow-oriented architecture that integrates various data processing tools, from ingestion to map visualization, using advanced Geographic Information System technologies and applying the architecture to support applications of the Parana's Institute of Rural Development (IDR-Paraná). This study aims to reduce human effort in map generation, provide updated data more quickly and accurately, and allow the implementation of new use cases with ease in the future.*

**Resumo.** *Atualmente, os dados são fontes extremamente valiosas para diversas áreas. Empresas, organizações e governos utilizam esses recursos para decisões estratégicas. Na agricultura, os dados podem gerar mapas essenciais para obter informações sobre fatores que impactam diretamente a produção agrícola. Este projeto propõe automatizar a geração de mapas, substituindo o trabalho manual por um processo automático, rápido e preciso. A proposta inclui a implementação de uma arquitetura orientada a fluxos que integra diversas ferramentas de processamento de dados, desde a ingestão até a visualização dos mapas, utilizando tecnologias avançadas de Sistema de Informação Geográfica e aplicando a arquitetura para auxiliar aplicações do Instituto de Desenvolvimento Rural do Paraná (IDR-Paraná). Espera-se reduzir o esforço humano na geração de mapas, fornecer dados atualizados de forma mais rápida e precisa e permitir a implementação de novos casos de uso futuros com facilidade.*

## 1. Introdução

A automatização de processos tem se tornado uma prática cada vez mais comum em diversas áreas, devido à sua capacidade de aumentar a eficiência, precisão e reduzir a necessidade de intervenção humana. No contexto da agricultura moderna, os mapas são ferramentas essenciais que fornecem dados climáticos vitais para a gestão das práticas agrícolas, informando várias questões essenciais para a agricultura, como a avaliação das condições que influenciam diretamente o crescimento das culturas, presença de doenças e pragas nos cultivares, distribuição de cultivares, entre outros [35, 7].

A relevância da automatização na produção desses mapas está na possibilidade de gerar informações precisas e atualizadas em tempo real, superando as limitações dos métodos manuais tradicionais que são demorados e suscetíveis a erros. A produção desses mapas pode ser feita de várias maneiras. Uma delas é através da utilização de *workflows*. Os *workflows* permitem incluir a reutilização de componentes e dados, a automação da coleta e integração de dados, e a aplicação de algoritmos de análise e visualização de dados. Esses processos automáticos reduzem o tempo e o esforço necessários, em contraste com um processo não automatizado, que está sujeito a ocorrência de falhas e dificuldade maior de atualizações e alterações de componentes do sistema.

Sendo assim, este projeto visa automatizar a geração de mapas utilizando uma arquitetura que integre diversas ferramentas de processamento de dados, desde a ingestão de dados até a visualização dos mapas. Obtendo isso, a aplicação será feita no IDR-Paraná, que servirá como estudo de caso nesse trabalho, uma vez que, atualmente, o Instituto realiza o processo manualmente. A utilização de uma arquitetura baseada em *workflows* é particularmente vantajosa, pois permite a integração eficiente de diferentes sistemas e ferramentas, garantindo que os dados sejam processados de forma rápida e eficaz.

Este documento está organizado da seguinte forma: a Seção 2 apresenta a fundamentação teórico-metodológica e o estado da arte, a Seção 3 detalha os objetivos deste trabalho, a Seção 4 descreve os procedimentos metodológicos, a Seção 5 apresenta o cronograma de execução, e a Seção 6 apresenta as contribuições e resultados esperados.

## **2. Fundamentação Teórico-Metodológica e Estado da Arte**

### **2.1. Produção de mapas agrometeorológicos**

A produção de mapas agrometeorológicos é uma prática essencial para a agricultura moderna, uma vez que através desse recurso é possível fornecer dados climáticos essenciais que coordenam as práticas agrícolas. Isso permite a avaliação das condições climáticas que influenciam diretamente o crescimento das culturas, a gestão da água e a análise de riscos relacionados com perigos climáticos [32].

Para produzir esses mapas, a obtenção de dados agrometeorológicos torna-se essencial. É possível obter esses dados através diferentes maneiras, que são elas:

- **Estações meteorológicas:** capazes de coletar temperatura, umidade, precipitação, radiação solar, velocidade e direção do vento, entre outras variáveis atmosféricas [27][26];
- **Sensoriamento remoto:** os satélites fornecem uma cobertura abrangente da superfície e da atmosfera da Terra e podem monitorar fenômenos de grande escala, como cobertura de nuvens, temperaturas da superfície do mar e mudanças na vegetação [3][4].

Em análises geoespaciais, é comum encontrar conjuntos de dados com valores conhecidos apenas em pontos específicos. Para criar uma representação contínua e precisa desses dados em toda a área de interesse, utiliza-se a interpolação, uma técnica essencial para estimar valores em locais onde não há amostras diretas. A precisão e a escolha do método de interpolação são cruciais para garantir a acurácia dos mapas gerados.

Interpolação é o processo de cálculo de valores desconhecidos a partir de valores conhecidos. Neste contexto, interpolação e extrapolação desempenham um papel vital. Por exemplo, um centro meteorológico mede a temperatura a cada hora do dia, mas pode ser necessário estimar a temperatura em intervalos de trinta minutos ou prever a temperatura para o dia seguinte. A previsão de valores intermediários dentro do intervalo de dados conhecidos é chamada de interpolação [37].

Vários trabalhos da literatura tiveram como objetivo identificar os métodos de interpolação mais adequados para diferentes variáveis climáticas. Para a temperatura do ar, os métodos que mais se destacaram foram a Krigagem e a Regressão, devido à sua capacidade de capturar as variações espaciais e fornecer estimativas precisas. No caso da precipitação, o método de Inverse Distance Weighting (IDW) mostrou o maior percentual de eficiência na estimativa de precipitação diária, proporcionando resultados mais confiáveis em comparação com outros métodos [28].

A produção de mapas envolve a utilização de diversas ferramentas e métodos, cada um com suas particularidades e vantagens específicas. Existem fundamentalmente duas abordagens de integração dessas ferramentas: a composição das suas funcionalidades em um Sistema de Informação Geográfica (SIG) e a integração de diferentes ferramentas em uma arquitetura fracamente acoplada. Essas abordagens serão melhor detalhadas nos tópicos a seguir.

## 2.2. Sistema de Informação Geográfica (SIG)

Um Sistema de Informação Geográfica (SIG), ou GIS (do inglês Geographic Information System), é uma ferramenta que permite a criação, geração, análise e mapeamento de diversos tipos de dados. Este sistema associa dados a um mapa, integrando informações de localização com informações descritivas. Isso proporciona uma base sólida para cartografia e análise, sendo amplamente utilizado na ciência e em quase todos os setores industriais. Os SIGs ajudam os usuários a entender padrões, relações e o contexto geográfico, oferecendo benefícios como melhor comunicação e eficiência, além de aprimorar a gestão e a tomada de decisões [10].

Esse sistema combina alguns componentes, essenciais para geração de mapas, sendo:

- **Hardware:** Dispositivos que tenham capacidade de armazenar grandes conjuntos de dados, softwares e aplicativos de SIG. Exemplos incluem servidores, computadores, tablets e drones equipados com GPS [8];
- **Software:** Programas, aplicativos, soluções que permitem executar funções essenciais de SIG, como manipulação, armazenamento, consulta e análise de dados e informações espaciais e não espaciais [8];
- **Dados:** Informações que podem ser espaciais (georreferenciadas) ou não espaciais. Dados espaciais incluem coordenadas geográficas, imagens de satélite e modelos digitais de elevação. Dados não espaciais incluem tabelas, relatórios e documentos que podem ser associados aos dados espaciais [8];
- **Métodos:** Diversas técnicas utilizadas para transformar dados em informações de fácil interpretação. Podem incluir algoritmos, fórmulas, modelos, estatísticas e outras técnicas de geoprocessamento [8];
- **Pessoas:** Envolvem tanto os profissionais que desenvolvem e utilizam soluções SIG, quanto o público que acessa mapas digitais [8].

Para gerar mapas, o SIG utiliza a combinação dos componentes mencionados. Inicialmente, os dados geoespaciais são coletados e armazenados no hardware adequado. O software de SIG, então, processa e analisa esses dados, aplicando métodos específicos de geoprocessamento para criar representações visuais. As pessoas, principalmente os profissionais de SIG, interpretam e refinam essas representações para produzir mapas precisos e úteis.

Apesar de suas amplas funcionalidades, o SIG apresenta algumas limitações. A criação de mapas pode exigir uma interação manual significativa, o que pode ser demorado e suscetível a erros humanos. Alternativamente, a automação através de códigos e scripts pode ser empregada, mas frequentemente esses scripts não são aproveitados de forma otimizada devido à complexidade e à necessidade de habilidades de programação especializadas. Essas limitações podem afetar a adaptabilidade ao sistema, tornando o processo mais lento e dependente, além de dificultar a usabilidade.

Tendo essas desvantagens em vista, implementar uma arquitetura integrada torna-se mais interessante para o objetivo deste trabalho, pois pode oferecer uma solução mais eficiente e acessível para a produção de mapas.

### **2.3. Arquiteturas integradas para a produção de mapas**

Arquiteturas integradas devem permitir que diferentes ferramentas troquem dados e funcionem em conjunto, melhorando a eficiência e produtividade, criando rotinas mais inteligentes e garantindo agilidade nos processos. A integração mantém os sistemas sincronizados, permitindo transferências de dados rápidas e adaptáveis. As ações são controladas por eventos, que disparam procedimentos ou solicitações.

Para estabelecer uma integração eficaz, é fundamental definir objetivos claros, mapear os processos internos e determinar quais integrações são necessárias [36]. As integrações de APIs facilitam o mapeamento e a transferência de dados, permitindo a exportação, agrupamento e análise dessas informações [33].

Diversos softwares e plataformas especializadas podem realizar integrações entre sistemas. Entre eles, destacam-se o IBM Cloud Pak for Integration e o Docker.

O IBM Cloud Pak for Integration é uma plataforma robusta que facilita a integração de aplicativos, dados e serviços em ambientes de TI complexos e híbridos. Ele combina várias ferramentas que permitem gerenciar APIs, integrar aplicativos, trocar mensagens e eventos, transferir dados de forma segura, automatizar fluxos de trabalho e monitorar essas integrações. No entanto, uma desvantagem dessa plataforma é que ela não é uma ferramenta de código aberto, o que pode resultar em custos adicionais de licenciamento e limitações na personalização, além de depender do suporte oficial da IBM [9].

Já o Docker, além de ser uma plataforma de containerização que permite empacotar aplicações e suas dependências em contêineres, garantindo consistência em diferentes ambientes e facilitando a integração, também possui a vantagem de ser uma ferramenta de código aberto. Isso a torna amplamente acessível e suportada por uma grande comunidade de desenvolvedores, promovendo a colaboração e a inovação contínua, além de reduzir custos com licenciamento [6].

As etapas fundamentais que compõem uma arquitetura integrada para a produção

de mapas serão detalhadas a seguir.

### 2.3.1. Ingestão de dados

A ingestão de dados é o processo de obter, importar e processar dados para uso posterior ou armazenamento em um banco de dados. Esse processo pode ser realizado manual ou automaticamente, utilizando uma combinação de ferramentas de software e hardware projetadas para essa finalidade [25].

O objetivo final da ingestão de dados é preparar os dados para uso imediato. Seja para análises, desenvolvimento de aplicativos ou aprendizado de máquina, o processo visa garantir que os dados sejam precisos, consistentes e prontos para utilização. Esse processo é essencial para a integração de dados, deduplicação, manutenção de restrições de integridade e carregamento em massa, frequentemente referido como Extrair, Transformar e Carregar (ETL) [18, 25].

A ingestão de dados pode ser feita em lote, onde os dados são processados de uma só vez, ou em tempo real, permitindo análises e ações imediatas [18]. Para essa etapa, destacam-se ferramentas como Apache Sqoop, ideal para transferências eficientes entre bancos de dados e Hadoop [20]; Apache Kafka, especializado em processamento de fluxo de dados em tempo real [24]; Apache Flume, voltado para a coleta e agregação de grandes volumes de dados de log [24]; e Apache NiFi, que oferece flexibilidade e controle no fluxo de dados entre sistemas [2]. Cada uma possui especificidades que atendem a diferentes necessidades de ingestão de dados.

### 2.3.2. Controle de fluxo de trabalho

Para finalmente automatizar a geração de mapas, é fundamental utilizar uma ferramenta capaz de orquestrar todos os fluxos de trabalho necessários e gerenciar pipelines de dados de maneira integrada. Um gerenciador de fluxo de trabalho é um sistema que administra processos e tarefas repetitivas em uma ordem específica. Ele coordena desde tarefas simples até processos de negócios complexos, envolvendo múltiplos fluxos de trabalho, sistemas de informação e dados. Geralmente, um fluxo de trabalho é visualizado em um diagrama para facilitar o planejamento e a execução repetitiva dos processos [19].

Existem dois tipos de fluxo de trabalho que podem ser adotados. São eles:

- **Grafo Acíclico Direcionado (DAG):** Nos DAGs, os nós representam tarefas e as arestas indicam dependências entre elas. Existem dependências de dados, onde a saída de uma tarefa é usada como entrada para outras, e dependências de controle, onde certas tarefas devem ser concluídas antes que outras possam começar. Um DAG pode representar um conjunto de programas em que a execução de um ou mais programas depende da conclusão bem-sucedida de outros [17];

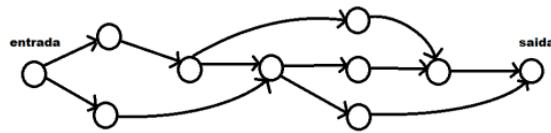


Figura 1. Exemplo de um Grafo Acíclico Direcionado (extraído de [5]).

- **Grafo Cíclico Direcionado (DCG):** Os fluxos baseados em DCG são caracterizados por ciclos que indicam loops, sejam implícitos ou explícitos, ou mecanismos de controle de iteração. Nesse contexto, o grafo de fluxo de trabalho descreve uma rede de tarefas, onde os nós representam processos, instâncias de componentes de software ou objetos de controle mais abstratos. Neste tipo de fluxo, os processos podem ser executados uma ou mais vezes, dependendo das condições estabelecidas [34].

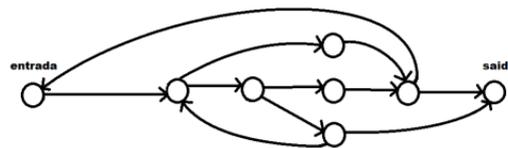


Figura 2. Exemplo de um Grafo Cíclico Direcionado (extraído de [5]).

Dentre as ferramentas que gerenciam fluxos, temos alguns exemplos como o Apache Airflow, Kedro e Luigi. O Apache Airflow é conhecido por sua flexibilidade e integração de sistemas diversos, ideal para *workflows* complexos [16, 11]. Kedro foca na estruturação e boas práticas de projetos de ciência de dados, promovendo modularidade e reprodutibilidade [21]. Luigi é uma opção eficaz para pipelines de dados em *batch*, valorizando a clareza na definição das tarefas [22]. Cada ferramenta se destaca em diferentes contextos e necessidades de projeto.

### 2.3.3. Geoprocessamento

O geoprocessamento envolve uma variedade de tarefas geográficas que requerem ferramentas específicas para manipulação, análise e visualização de dados espaciais. Essas ferramentas podem incluir bibliotecas de software, sistemas de informação geográfica (SIG), e servidores de mapas.

Algumas bibliotecas existentes que atendem essas tarefas são:

- **GDAL:** Biblioteca de código aberto que fornece suporte para leitura e escrita de diferentes formatos de dados geoespaciais. Através dela, é possível realizar conversão de formatos, manipulação de dados raster e vetoriais, re-projeção e geoprocessamento. É utilizada em muitos softwares de geoprocessamento para garantir compatibilidade com uma ampla variedade de formatos de dados geoespaciais [12];

- **Shapely:** Biblioteca de manipulação geométrica baseada nas interfaces da biblioteca GEOS. Permite realizar criação, análise e manipulação de formas geométricas bidimensionais e é muito utilizada em aplicações Python para operações de geoprocessamento e análise espacial [31];
- **Rasterio:** Biblioteca Python que facilita a leitura e a escrita de dados raster geoespaciais, manipulação, recorte e amostragem desses dados. É ideal para trabalhos que envolvem dados de imagens de satélites e outros dados raster [30];
- **GeoPandas:** Extensão da biblioteca pandas que facilita o trabalho com dados geoespaciais vetoriais em Python. Oferece suporte para operações geoespaciais, como intersecções, união e diferenciação de formas geométricas, tudo integrado com a funcionalidade de pandas. Usada para análise espacial de dados vetoriais, integração com sistemas de informação geográfica e visualização de dados geoespaciais [13];
- **PyKrig:** Biblioteca Python para krigagem, um método de interpolação geoespacial, é capaz de realizar várias formas de krigagem, incluindo krigagem ordinária, universal e cokrigagem. É utilizada em geociências para interpolação e análise espacial de dados geoespaciais contínuos [29].

Agora, dentre os servidores de mapas, temos:

- **GeoServer:** Servidor de mapas de código aberto que permite a publicação de dados espaciais como serviços web e disponibiliza suporte a padrões OGC (Open Geospatial Consortium) como WMS (Web Map Service) e WFS (Web Feature Service). É utilizado para criar serviços de mapas interativos e integrar dados espaciais em aplicativos web [14];
- **MapServer:** Semelhante ao GeoServer, é uma plataforma de código aberto para publicação de dados espaciais e criação de aplicações de mapeamento na web. Permite a renderização de mapas dinâmicos, suporte a uma variedade de formatos de dados e padrões OGC. É usado para desenvolver aplicações web geoespaciais e publicar dados espaciais online [23].

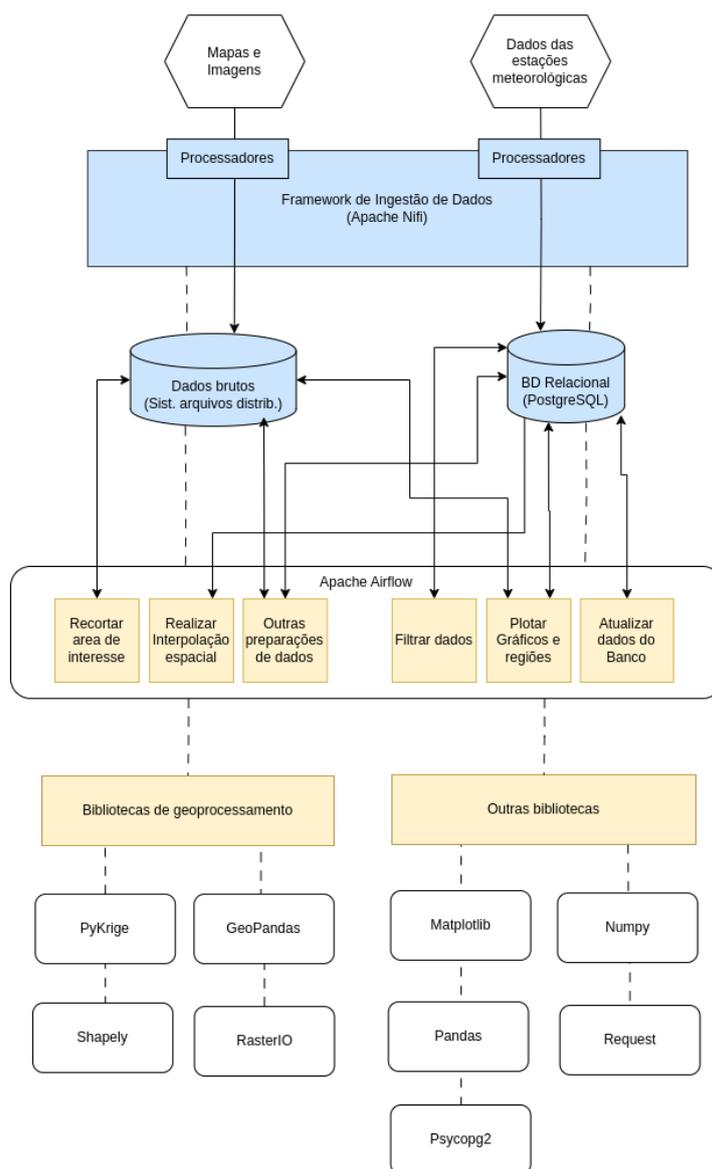
Essa foi uma breve revisão sobre as opções disponíveis de ferramentas. Foi realizado um estudo mais aprofundado em um Trabalho de Conclusão de Curso anterior a esse [5], que servirá como base a partir de agora. Portanto, o próximo tópico irá detalhar a arquitetura escolhida, que foi estudada nesse trabalho anterior.

## 2.4. Arquitetura escolhida

A arquitetura escolhida para este projeto foi desenhada para atender a três etapas principais já mencionadas anteriormente: ingestão de dados, orquestração de tarefas e geoprocessamento. As ferramentas e bibliotecas foram selecionadas com base em suas funcionalidades e desempenho para cada etapa do processo, desde a ingestão de dados até a apresentação dos resultados.

- Para a **comunicação** entre todas as ferramentas da arquitetura, a melhor opção é utilizar contêineres do Docker, por ser uma ferramenta de código aberto e gratuita, além de permitir fácil gerenciamento de dependências e simplificar o processo de implantação;
- Para a **ingestão de dados** a escolha foi o Apache NiFi, pela sua capacidade de trabalhar com dados em lote e dados em tempo real, além da facilidade de uso, robustez e escalabilidade;

- Para o **controle de fluxo de trabalho**, o Apache Airflow foi o que mais se enquadrou nos parâmetros, por permitir a criação, monitoramento e agendamento de fluxos de trabalho, além de sua flexibilidade e escalabilidade dos processos;
- Para o **geoprocessamento**, foram utilizadas várias bibliotecas Python, incluindo GeoPandas, PyKrige, Rasterio e Shapely. Além dessas, outras bibliotecas Python, como Pandas, Matplotlib e NumPy, são utilizadas para tratamento de dados não espaciais, criação de gráficos e manipulação de *arrays*.



**Figura 3. Esquema que representa a arquitetura adotada para gerenciar o fluxo de trabalho da geração de mapas (extraído de [5]).**

Tendo em mãos a arquitetura, é possível trabalhar em cima da resolução dos problemas ainda existentes: o IDR-Paraná realiza a geração de mapas de forma manual, o

que torna o processo dependente de um recurso humano para a sua funcionalidade, o que também não permite a disponibilização dos dados em tempo real. Além disso, no trabalho anterior foram feitos alguns estudos de caso que são iniciais, então é preciso aplicar a arquitetura em problemas práticos e colocar em produção, para finalmente disponibilizar o serviço.

### 3. Objetivos

O objetivo geral deste projeto é automatizar o processo de geração de mapas agrometeorológicos utilizando uma arquitetura de ferramentas de processamento de grandes volumes de dados orientada a fluxos.

Os objetivos específicos são:

- Definir um conjunto de tarefas envolvidas no processo de geração de mapas que possam ser reutilizadas entre diferentes mapas;
- Construir *pipelines* para automatização do processo de criação de mapas, em especial a partir de dados de estações meteorológicas e de ocorrências de eventos de interesse fornecidos pelo IDR-Paraná, encadeando tarefas implementadas sobre a arquitetura utilizada;
- Integrar um servidor de mapas à arquitetura utilizada, com suporte à visualização a partir de mapas estáticos exportados como imagem e mapas dinâmicos interativos utilizando bibliotecas de *web maps*.

### 4. Procedimentos metodológicos/Métodos e técnicas

Para a realização deste trabalho, será necessário primeiro um estudo aprofundado sobre o tema. Isso inclui compreender como funcionam as ferramentas e bibliotecas que serão utilizadas, bem como aprender a trabalhar com dados geoespaciais e métodos envolvidos nesta atividade, tais como algoritmos de interpolação para geração de mapas climáticos e outros tratamentos. Esse estudo preliminar é crucial para garantir a correta aplicação das técnicas e a confiabilidade dos resultados.

Durante a fase inicial do estudo, serão definidos os casos de uso específicos a serem implementados no sistema. Esta fase será conduzida com base em *sprints* iterativos e incrementais, permitindo um planejamento ágil e adaptativo. Cada *sprint* focará na conceituação detalhada de um caso de uso por vez, evitando a necessidade imediata de coletar a base de dados completa para todos os casos. Esta abordagem permitirá uma atenção concentrada em cada caso de uso, facilitando a identificação e a resolução de problemas específicos de forma mais eficaz.

Após a definição dos casos de uso, o próximo passo será a preparação do ambiente de trabalho. Isso inclui a instalação e teste de funcionalidade de todos os softwares e ferramentas que serão utilizados ao longo do processo. A arquitetura proposta no trabalho que antecede este será complementada com novos módulos, que focarão na implementação do servidor de mapas e na integração de bibliotecas de *web maps*.

Esses módulos incluirão novas ferramentas, como:

- **Leaflet:** Biblioteca JavaScript para visualização e interatividade de mapas no frontend, proporcionando uma interface dinâmica e responsiva para os usuários [1];

- **GeoServer:** Servidor de mapas que gerencia a publicação, processamento e fornecimento de dados geoespaciais no backend, garantindo eficiência e escalabilidade na manipulação de grandes volumes de dados [15].

A escolha dessas ferramentas se deve à utilização das duas em conjunto, pois ao serem combinadas oferecem uma solução completa para o desenvolvimento de aplicações de mapeamento interativo e análise geoespacial. A biblioteca GeoServer, apesar de ser mais focada em mapas dinâmicos, permite também o gerenciamento de mapas estáticos, poupando utilização de mais servidores.

Com o ambiente preparado, a ingestão de dados será iniciada. Utilizando o Apache NiFi, será possível automatizar o fluxo de dados, coletando-os de diversas fontes, como bancos de dados públicos, APIs e arquivos locais. Esse processo garantirá que os dados sejam recebidos de forma contínua e organizada.

Em seguida, será realizado o tratamento desses dados, visto que os dados brutos geralmente contêm ruídos e inconsistências que precisam ser corrigidos. A aplicação de técnicas de limpeza como remoção de valores nulos, correção de erros e padronização será feita nessa etapa.

Após o tratamento, os dados serão utilizados para produzir mapas no formato raster. A utilização de métodos de interpolação será essencial nessa etapa, pois possibilitará a criação de superfícies contínuas e precisas a partir de amostras pontuais. No entanto, como existem vários métodos de interpolação, é fundamental analisar cada método para determinar qual se adequa melhor aos casos de uso que serão abordados no trabalho.

Em seguida, será utilizada a ferramenta Apache Airflow para programar e monitorar pipelines de processamento e análise dos dados. Isso permitirá a automação de tarefas repetitivas e a garantia de que os processos sejam executados de maneira eficiente e ordenada.

A visualização dos resultados será realizada utilizando as bibliotecas de geoprocessamento citadas na arquitetura, a fim de gerar a preparação visual dos mapas com a escolha adequada de cores, legendas e escalas, garantindo que as informações sejam claras e acessíveis.

Finalmente, para disponibilizar os mapas gerados de forma interativa e acessível, será configurado um servidor de mapas utilizando os novos módulos citados anteriormente. Esses módulos permitirão a publicação e a visualização dos mapas em um ambiente web, permitindo o acesso aos usuários.

## 5. Cronograma de Execução

Nesta seção, serão listadas as atividades descritas na seção anterior. Também é apresentado na tabela 1 a seguir o cronograma de execução das atividades.

1. Revisão bibliográfica focada nos conceitos e ferramentas necessários e definição dos casos de uso;
2. Preparação do ambiente de trabalho e realização de testes de funcionalidades das ferramentas escolhidas;
3. Ingestão de dados;
4. Tratamento dos dados;

5. Geração de mapas do primeiro caso de uso;
6. Programação e Monitoramento de Pipelines;
7. Visualização e Apresentação dos resultados;
8. Configuração de um servidor de mapas;
9. Busca da base de dados dos casos de uso restantes;
10. Implementação e teste dos casos de uso restantes;
11. Escrita do TCC (versão preliminar);
12. Escrita do TCC (versão para a banca examinadora).

Atividade	2024							2025	
	Jun	Jul	Ago	Set	Out	Nov	Dez	Jan	Fev
1	•	•	•						
2		•	•						
3			•						
4			•	•					
5				•					
6					•				
7					•				
8						•			
9						•			
10						•	•	•	
11		•	•	•	•				
12						•	•	•	•

**Tabela 1. Cronograma de execução**

## 6. Contribuições e/ou Resultados esperados

Espera-se que este trabalho resulte na geração automatizada de mapas, capazes de fornecer diversas informações com apenas pequenas alterações. A automatização da geração dos mapas pode contribuir com o IDR para reduzir o esforço na geração dos mapas, uma vez que atualmente isso tem sido feito manualmente. Além disso, a automatização também permite a disponibilização dos dados de forma mais rápida, já que dispensa a necessidade de atualização manual. A utilização de um fluxo de tarefas facilita a implementação de novos casos de uso no futuro, caso o Instituto almeje. Com apenas algumas novas implementações, o uso das ferramentas da arquitetura torna o processo mais intuitivo e compreensível.

## 7. Espaço para assinaturas

Londrina, 29 de julho de 2024.

*Batiz Bonino Caporosso*

Aluno

Orientador

## Referências

- [1] Volodymyr Agafonkin. Leaflet overview, 2024. Disponível em: <https://leafletjs.com/>. Acesso em: 24 jul 2024.
- [2] APACHE. Apache nifi overview, 2023. Disponível em: <https://nifi.apache.org/docs/nifi-docs/html/overview.html>. Acesso em: 24 jul 2024.
- [3] Gustavo Camps-Valls, Devis Tuia, Luis Gómez-Chova, Sandra Jiménez, and Jesús Malo. *Remote sensing image processing*, volume 12. Morgan and Claypool Publishers, 12 2012.
- [4] Emilio Chuvieco. *Fundamentals of satellite remote sensing : an environmental approach*. Taylor & Francis, 2016.
- [5] Jennifer do Prado da Silva. Proposta de uma arquitetura de gerência de dados para tarefas de aprendizado de máquina e análise de dados na agricultura. Trabalho de Conclusão de Curso (Graduação em Ciência da Computação) – Universidade Estadual de Londrina, Paraná, 2024.
- [6] Docker. Docker overview, 2024. Disponível em: <https://docs.docker.com/guides/docker-overview/>. Acesso em: 24 jul 2024.
- [7] Andrei Dornik, Marinela Adriana Chețan, Lucian Drăguț, Andrei Iliuță, and Daniel Dorin Dicu. Importance of the mapping unit on the land suitability assessment for agriculture. *Computers and Electronics in Agriculture*, 201:107305, 2022.
- [8] Infosys BPM Dr. Pradeep Kishore. Geospatial data services, 2024. Disponível em: <https://www.infosysbpm.com/blogs/geospatial-data-services/gis-five-essential-components.html>. Acesso em: 24 jul 2024.
- [9] IBM Cloud Education. What is an integration platform? do i need one?, 2021. Disponível em: <https://www.ibm.com/blog/integration-platform/>. Acesso em: 24 jul 2024.
- [10] Esri. O que é gis?, 2024. Disponível em: <https://www.img.com.br/pt-br/o-que-e-gis/visao-geral>. Acesso em: 24 jul 2024.
- [11] Leanne Finnigan and Emily Toner. Building and maintaining metadata aggregation workflows using apache airflow. *Temple University Libraries*, 2021.
- [12] GDAL. Gdal, 2024. Disponível em: <https://gdal.org/#>. Acesso em: 24 jul 2024.
- [13] GeoPandas. Geopandas 1.0.1, 2013. Disponível em: <https://geopandas.org/en/stable/about.html>. Acesso em: 24 jul 2024.
- [14] GeoServer. Geoserver, 2024. Disponível em: <https://geoserver.org/>. Acesso em: 24 jul 2024.
- [15] GeoServer. What is geoserver?, 2024. Disponível em: <https://geoserver.org/about/#:~:text=GeoServer%20is%20a%20Java%20Dbased,map%20creation%20and%20data%20sharing..> Acesso em: 24 jul 2024.
- [16] Bas P Harenslak and Julian de Ruiter. *Data pipelines with apache airflow*. Simon and Schuster, 2021.

- [17] Jiahao He, Guangju Wang, Guangyuan Zhang, and Jiheng Zhang. Consensus mechanism design based on structured directed acyclic graphs. *Blockchain: Research and Applications*, 2(1):100011, 2021.
- [18] IBM. Complete guide to data ingestion: Types, process, and best practices, 2023. Disponível em: <https://www.ibm.com/blog/guide-to-data-ingestion/>. Acesso em: 24 jul 2024.
- [19] IBM. What is a workflow?, 2023. Disponível em: <https://www.ibm.com/topics/workflow>. Acesso em: 24 jul 2024.
- [20] Jarek Jarcec Cecho Kathleen Ting. *Apache sqoop cookbook: Unlocking hadoop for your relational database*. O'Reilly Media, 2013.
- [21] Kedro. Introduction to kedro, 2024. Disponível em: <https://docs.kedro.org/en/stable/introduction/index.html>. Acesso em: 24 jul 2024.
- [22] Luigi. Introduction to luigi, 2023. Disponível em: <https://luigi.readthedocs.io/en/stable/>. Acesso em: 24 jul 2024.
- [23] MapServer. Welcome to mapserver, 2024. Disponível em: <https://mapserver.org/>. Acesso em: 24 jul 2024.
- [24] Andreea Mătăcuță and Cătălina Popa. Big data analytics: Analysis of features and performance of big data ingestion tools. *Informatica Economica*, 22(2), 2018.
- [25] John Meehan, Cansu Aslantas, Stan Zdonik, Nesime Tatbul, and Jiang Du. Data ingestion for the connected world. In *Cidr*, volume 17, pages 8–11, 2017.
- [26] BM Mote and DD Sahu. *Principles of Agricultural Meteorology*. Scientific Publishers, 2014.
- [27] Bernhard Pacher, Branislava Lalic, Josef Eitzinger, Anna Dalla Marta, Simone Orlandini, and Ana Firanj Sremac. *Agricultural meteorology and climatology*. Firenze University Press, 2018.
- [28] Edenir Bagio PERIN, Luiz Fernando de Novaes VIANNA, Wilian da Silva RICCE, Angelo Mendes MASSIGNAM, and Cristina Pandolfo. Interpolação das variáveis climáticas temperatura do ar e precipitação: Revisão dos métodos mais eficientes, 2015.
- [29] PyKrige. Pykrige, 2024. Disponível em: <https://geostat-framework.readthedocs.io/projects/pykrige/en/stable/>. Acesso em: 24 jul 2024.
- [30] Rasterio. Rasterio: access to geospatial raster data, 2018. Disponível em: <https://rasterio.readthedocs.io/en/stable/>. Acesso em: 24 jul 2024.
- [31] Shapely. The shapely user manual, 2024. Disponível em: <https://shapely.readthedocs.io/en/stable/manual.html>. Acesso em: 24 jul 2024.
- [32] Kees Stigter. *Applied agrometeorology*. Springer Berlin Heidelberg, 2010.
- [33] SYDLE. Systems integration: Learn the kinds, challenges, and importance, 2022. Disponível em: <https://www.sydle.com/blog/systems-integration-6140d39a84679b13bf127a93>. Acesso em: 24 jul 2024.

- [34] Domenico Talia. Workflow systems for science: Concepts and tools. *International Scholarly Research Notices*, 2013(1):404525, 2013.
- [35] Big Tires. Como o mapeamento agrícola auxilia na produtividade do solo?, 2022. Disponível em: <https://www.bigtires.com.br/blog/post/vantagens-mapeamento-agricola>. Acesso em: 24 jul 2024.
- [36] Iwan van Beurden. The meaning of tool integration, 2016. Disponível em: <https://www.exida.com/blog/the-meaning-of-tool-integration>. Acesso em: 24 jul 2024.
- [37] Muhammad Abdul Wahab. Interpolation and extrapolation. *Proc. Topics Syst. Eng. Winter Term*, 17:1–6, 2017.