



UNIVERSIDADE FEDERAL DO PARÁ
CAMPUS UNIVERSITÁRIO DE ANANINDEUA
FACULDADE DE TECNOLOGIA EM GEOPROCESSAMENTO

JOÃO GUILHERME SILVA RIBEIRO DOS SANTOS
SÁVIO CARDOSO DE SOUZA LEAL

**DESENVOLVIMENTO DE FERRAMENTAS AUTOMATIZADAS EM PYTHON
PARA OTIMIZAÇÃO DE PROCESSOS NO QGIS COM INTERFACE AO
APLICATIVO SEA-GEOAMAZONIA**

ANANINDEUA, PA

2025

UNIVERSIDADE FEDERAL DO PARÁ
CAMPUS UNIVERSITÁRIO DE ANANINDEUA
FACULDADE DE TECNOLOGIA EM GEOPROCESSAMENTO

JOÃO GUILHERME SILVA RIBEIRO DOS SANTOS
SÁVIO CARDOSOS DE SOUZA LEAL

**DESENVOLVIMENTO DE FERRAMENTAS AUTOMATIZADAS EM PYTHON
PARA OTIMIZAÇÃO DE PROCESSOS NO QGIS COM INTERFACE AO
APLICATIVO SEA-GEOAMAZONIA**

Trabalho de Curso apresentado para obtenção do grau de Tecnólogo(a) em Geoprocessamento pela Faculdade de Tecnologia em Geoprocessamento da Universidade Federal do Pará, Campus Ananindeua.

Orientador: Prof. Dr. Marcelo Augusto Machado Vasconcelos

ANANINDEUA, PA

2025

AUTORIZO A REPRODUÇÃO E DIVULGAÇÃO TOTAL OU PARCIAL DESTES
TRABALHO, POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO, PARA
FINS DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

**Dados Internacionais de Catalogação na Publicação (CIP) de acordo com ISBD
Sistema de Bibliotecas da Universidade Federal do Pará
Gerada automaticamente pelo módulo Ficat, mediante os dados fornecidos pelo(a) autor(a)**

S586d Silva Ribeiro dos Santos, João Guilherme.
DESENVOLVIMENTO DE FERRAMENTAS
AUTOMATIZADAS EM PYTHON PARA OTIMIZAÇÃO DE
PROCESSOS NO QGIS COM INTERFACE AO APLICATIVO
SEA-GEOAMAZONIA / João Guilherme Silva Ribeiro dos
Santos, Sávio Cardoso de Souza. — 2025.
72 f. : il. color.

Orientador(a): Prof. Dr. Marcelo Augusto Machado
Vasconcelos
Trabalho de Conclusão (Graduação) - Universidade Federal do
Pará, Campus Universitário de Ananindeua, Curso de
Geoprocessamento, Ananindeua, 2025.

1. Automação. 2. Qgis. 3. SIRGAS 2000. 4. SEA-
GEOAMAZONIA. 5. GEOPROCESSAMENTO. I. Título.

CDD 005

JOÃO GUILHERME SILVA RIBEIRO DOS SANTOS
SÁVIO CARDOSOS DE SOUZA LEAL

**DESENVOLVIMENTO DE FERRAMENTAS AUTOMATIZADAS EM PYTHON
PARA OTIMIZAÇÃO DE PROCESSOS NO QGIS COM INTERFACE AO
APLICATIVO SEA-GEOAMAZONIA**

Trabalho de Curso apresentado para obtenção do grau de Tecnólogo(a) em Geoprocessamento pela Faculdade de Tecnologia em Geoprocessamento da Universidade Federal do Pará, Campus Ananindeua.

Data de aprovação: 27/03/2025

Conceito: Excelente

Ananindeua – PA

Orientador: Prof. Dr. Marcelo Augusto Machado Vasconcelos
UFPA – Universidade Federal Do Pará

Prof. Dr. Artur Vinicius Ferreira dos Santos
UFPA – Universidade Federal Do Pará

João Guilherme Silva Ribeiro dos Santos

Sávio Cardoso De Souza Leal

ANANINDEUA, PARÁ

2025

Dedico este trabalho às nossas famílias, pelo apoio constante, pela motivação nos momentos difíceis e por acreditarem em nosso potencial ao longo de toda essa jornada.

AGRADECIMENTOS

Ao Deus Todo-Poderoso, nosso eterno guia, dedicamos nosso mais profundo reconhecimento. Foi em Sua presença que encontramos força, coragem e sabedoria para superar os desafios que surgiram ao longo desta jornada. A Ele confiamos nossos sonhos e, por Sua infinita bondade, recebemos as bênçãos necessárias para seguir firmes até a conclusão deste trabalho. Nada disso seria possível sem a fé que nos sustentou e sem a certeza de que Seus propósitos sempre se cumprem no tempo certo.

A nossa família, fonte de amor, inspiração e apoio incondicional, expressamos nossa eterna gratidão. Cada palavra de incentivo, cada gesto de cuidado, cada momento de compreensão foi essencial para que pudéssemos chegar até aqui. A vocês, que acompanharam de perto nossas lutas e vitórias, dedicamos não apenas este trabalho, mas todo o esforço nele investido. Esta conquista também é de vocês, que sempre acreditaram em nosso potencial e nos incentivaram a seguir adiante.

Aos professores e orientadores que fizeram parte da nossa formação acadêmica, registramos nossa sincera admiração e respeito. Suas orientações, ensinamentos e dedicação foram fundamentais para o nosso crescimento intelectual e pessoal. A cada aula, a cada conversa, levaremos conosco o aprendizado transmitido com tanto zelo e compromisso com a educação.

Aos colegas de curso e amigos que compartilharam conosco os desafios desta trajetória, deixamos nossa gratidão pelas trocas, pelas experiências vividas e pelo apoio mútuo ao longo dos anos. A jornada foi mais leve e significativa por termos caminhado juntos.

"A tecnologia move o mundo."

— **Steve Jobs**

RESUMO

O presente Trabalho de Conclusão de Curso apresenta a criação e implementação de uma metodologia automatizada no software QGIS, destinada a análises regressivas e geoespaciais, focadas na conversão de geometrias para o sistema de coordenadas SIRGAS 2000/UTM e no desenvolvimento de layouts padronizados para mapeamento temático. O estudo propõe soluções práticas para automatizar processos de análise espacial, como o cálculo de áreas e perímetros, a conversão precisa para zonas UTM e a adaptação de simbologias temáticas em camadas geográficas. Utilizou-se uma abordagem baseada em scripts Python integrados ao QGIS, com ênfase em otimizar a personalização de layouts cartográficos e gerar relatórios detalhados baseados em atributos das camadas de dados. A metodologia foi aplicada a um conjunto de dados reais, possibilitando a avaliação dos resultados gerados e a validação das ferramentas desenvolvidas. O sistema desenvolvido foi implementado com interface no SEA-GEOAMAZONIA, permitindo a integração e análise espacial de dados geográficos em larga escala. Essa integração possibilitou maior eficiência no processamento de informações territoriais e na padronização dos layouts cartográficos. Como resultado, foi obtido um sistema que facilita a criação de mapas com alta precisão e análise eficiente dos dados espaciais, respeitando os padrões cartográficos e as necessidades de ajustes personalizados pelos usuários. As conclusões indicam que o desenvolvimento de ferramentas automatizadas em ambientes SIG pode reduzir o tempo de execução de tarefas complexas, aumentar a precisão dos resultados e fornecer subsídios mais claros para a tomada de decisão em projetos geoespaciais.

Palavras-Chave: Automação, QGIS, SIRGAS 2000, Geoprocessamento, Análise Espacial, SEA-GEOAMAZONIA.

ABSTRACT

This Final Paper presents the creation and implementation of an automated methodology in QGIS software, designed for regressive and geospatial analyses, focusing on the conversion of geometries to the SIRGAS 2000/UTM coordinate system and the development of standardized layouts for thematic mapping. The study proposes practical solutions to automate spatial analysis processes, such as calculating areas and perimeters, accurately converting to UTM zones, and adapting thematic symbologies in geographic layers. A script-based approach using Python integrated into QGIS was adopted, emphasizing the optimization of cartographic layout customization and the generation of detailed reports based on layer attributes. The methodology was applied to a real dataset, allowing for an evaluation of the generated results and the validation of the developed tools. The system was implemented with an interface in SEA-GEOAMAZONIA, enabling the integration and spatial analysis of large-scale geographic data. This integration improved the efficiency of territorial data processing and the standardization of cartographic layouts. As a result, a system was developed that facilitates the creation of highly accurate maps and efficient spatial data analysis, adhering to cartographic standards and users' customization needs. The conclusions indicate that the development of automated tools in GIS environments can reduce the execution time of complex tasks, increase result accuracy, and provide clearer support for decision-making in geospatial projects.

Keywords: Automation, QGIS, SIRGAS 2000, Geoprocessing, Spatial Analysis, SEA-GEOAMAZONIA.

LISTA DE FIGURAS

| | |
|---|----|
| Figura 1– Página Inicial do QGIS..... | 21 |
| Figura 2-Demonstração de Localização do Terminal Python. | 21 |
| Figura 3 - Demonstração do Terminal Python Aberto..... | 22 |
| Figura 4 - Demonstração do Terminal Python com a Função Mostrar Editor Aberta..... | 22 |
| Figura 5 - Demonstração do Botão de Execução do Script..... | 23 |
| Figura 6 – Código Completo de Identificação de Zona UTM de um Shapefile no Qgis..... | 26 |
| Figura 7 - Demonstração dos Estados Usados para o Teste..... | 26 |
| Figura 8 - Resultados do Funcionamento do Código de Identificação das Zonas UTM. | 27 |
| Figura 9 - Demonstração Completa de Cálculo de Área e Perímetro – Parte 1 | 29 |
| Figura 10 - Demonstração Completa de Cálculo de Área e Perímetro – Parte 2 | 30 |
| Figura 11 - Resultado do Funcionamento do Código de Cálculo de Área e Perímetro..... | 30 |
| Figura 12 - Organização das Camadas..... | 33 |
| Figura 13 - Condições no Gerenciador de Layout..... | 34 |
| Figura 14 - Condições no Layout para o Funcionamento do Código | 34 |
| Figura 15 - Código Completo de Automação de um Mapa (Parte 1). | 35 |
| Figura 16 - Código Completo de Automação de um Mapa (Parte 2). | 35 |
| Figura 17 - Código Completo de Automação de um Mapa (Parte 3). | 36 |
| Figura 18 - Código Completo de Automação de um Mapa (Parte 4). | 36 |
| Figura 19 - Código Completo de Automação de um Mapa (Parte 5). | 37 |
| Figura 20 - Configure esse Trecho para Adaptabilidade para o Funcionamento do Código. .. | 37 |
| Figura 21 - Demonstração das Camadas e Grupos Criados para o Teste. | 38 |
| Figura 22 - Ajuste Realizado para as Camadas em Questão para serem usadas. | 39 |
| Figura 23 - Demonstração da Confirmação do Funcionamento do Código..... | 40 |
| Figura 24 - Confirmação do Layout Criado no Gerenciador. | 40 |
| Figura 25 - Resultado final e Criação do Layout Adaptado para as Camadas Escolhidas. | 41 |
| Figura 26 - Bloco das Bibliotecas para o Funcionamento do Código do SEA GEO. | 44 |
| Figura 27 - Função para Ajustar a Extensão do Mapa no SEA GEO. | 45 |
| Figura 28 - Função para Dimensão do Mapa no SEA GEO. | 45 |
| Figura 29 - Função de Altura e Comprimento do Mapa SEA GEO..... | 46 |
| Figura 30 - Funções para Gerar o Mapa SEA GEO – Parte 1..... | 47 |
| Figura 31 - Funções para Gerar o Mapa SEA GEO – Parte 2..... | 47 |
| Figura 32 - Configuração do Usuário no Mapa SEA GEO..... | 49 |

| | |
|--|----|
| Figura 33 - Formação dos Grupos para o Funcionamento do Mapa SEA GEO. | 51 |
| Figura 34 - Código Completo do Mapa SEA GEO – Parte 1..... | 52 |
| Figura 35 - Código Completo do Mapa SEA GEO – Parte 2..... | 52 |
| Figura 36 - Código Completo do Mapa SEA GEO – Parte 3..... | 53 |
| Figura 37 - Código Completo do Mapa SEA GEO – Parte 4..... | 53 |
| Figura 38 - Verificação dos Grupos "SEA" e "MAPA DE LOCALIZAÇÃO - SEA"..... | 54 |
| Figura 39 - Criação das Colunas na Tabela de Atributo da Camada. | 55 |
| Figura 40 - Verificação da Renomeação Está Correta..... | 55 |
| Figura 41 - Verificação se o "Mapa_SEA_GEO" Foi Criada. | 56 |
| Figura 42 - Mapa Completo com o Modelo SEA GEO..... | 57 |
| Figura 43 - Gerenciamento de Layout Contendo o Relatório. | 57 |
| Figura 44 - Temas Separados Para o Relatório. | 58 |
| Figura 45 - Atualização do Tema Para Cada Relatório. | 59 |
| Figura 46 - Salvamento do Relatório em PDF. | 59 |
| Figura 47 - Relatório Finalizado em PDF. | 60 |
| Figura 48 - Apresentação do Cabeçalho e Rodapé..... | 60 |
| Figura 49 - Representação das Seções da Aba Relatório..... | 61 |
| Figura 50 - Tabela de Estudo do SEA GEO no Estudo de Caso em Quintais Produtivos na Ilha de Caratateua. | 67 |
| Figura 51 - Exemplo de Gráfico do SEA GEO no Estudo de Caso em Quintais Produtivos na Ilha de Caratateua. | 67 |
| Figura 52 - Tabela do SEA GEO na Análise Regressiva de Sistemas Agroecológicos. | 69 |
| Figura 53 - Gráfico do SEA GEO na Análise Regressiva de Sistemas Agroecológicos. | 69 |

LISTA DE ABREVIATURAS

| | |
|----------------|--|
| API | Application Programming Interface (Interface de Programação de Aplicações) |
| CRS | Coordinate Reference System (Sistema de Referência de Coordenadas) |
| DBMS | Database Management System (Sistema de Gerenciamento de Banco de Dados) |
| EPSG | European Petroleum Survey Group (Código de Sistemas de Referência) |
| GDAL | Geospatial Data Abstraction Library |
| GIS | Geographic Information System (Sistema de Informação Geográfica) |
| GUI | Graphical User Interface (Interface Gráfica do Usuário) |
| IDE | Integrated Development Environment (Ambiente de Desenvolvimento Integrado) |
| OGR | OpenGIS Simple Features Reference Implementation |
| OS | Operating System (Sistema Operacional) |
| PDF | Portable Document Format |
| PNG | Portable Network Graphics |
| PROJ | Biblioteca de Projeções Cartográficas |
| PyQGIS | Python para QGIS |
| QGIS | Quantum GIS (Sistema de Informação Geográfica de Código Aberto) |
| RAM | Random Access Memory (Memória de Acesso Aleatório) |
| SIRGAS 2000 | Sistema de Referência Geocêntrico para as Américas |
| SEA-GEO | Sistema de Avaliação Espacial e Geográfica da Amazônia |
| SIG | Sistema de Informação Geográfica |
| SQL | Structured Query Language (Linguagem de Consulta Estruturada) |
| UTM | Universal Transverse Mercator (Sistema de Coordenadas Universais Transversais de Mercator) |

SUMÁRIO

| | | |
|-------|---|----|
| 1 | INTRODUÇÃO | 15 |
| 1.1 | Objetivos | 16 |
| 1.2 | Estrutura Do Trabalho..... | 16 |
| 2 | CONFIGURAÇÃO E AMBIENTE DE DESENVOLVIMENTO NO QGIS | 18 |
| 3 | CONFIGURAÇÃO DO SISTEMA E HARDWARE UTILIZADO | 19 |
| 4 | DIRETRIZES DE IMPLEMENTAÇÃO E DESENVOLVIMENTO DE CÓDIGOS PADRÃO | 20 |
| 4.1 | Passos Fundamentais Para Garantir A Execução Adequada Dos Códigos..... | 20 |
| 5 | INSTRUÇÕES E EXEMPLOS DE CÓDIGOS PARA AUTOMAÇÃO DE PROCESSOS NO QGIS: GUIA COMPLETO PARA IMPLEMENTAÇÃO E EXECUÇÃO .. | 24 |
| 5.1 | Explicação Do Código: Identificador De Zona Utm De Um Shapefile No Qgis..... | 24 |
| 5.1.1 | Código: Identificação De Zona Utm De Um Shapefile No Qgis (Anexo 1) (Figura 6). | 26 |
| 5.1.2 | Testes Do Código: Identificação De Zona Utm De Um Shapefile No Qgis..... | 26 |
| 5.2 | Explicação Do Código: Conversão De Camada Para Utm Em Sirgas 2000 E Cálculo De Área E Perímetro No Qgis | 27 |
| 5.2.1 | Código: Conversão De Camada Para Utm Em Sirgas 2000 E Cálculo De Área E Perímetro No Qgis (Anexo 2) (Figura 9) (Figura 10)..... | 29 |
| 5.2.2 | Testes Do Código: Conversão De Camada Para Utm Em Sirgas 2000 E Cálculo De Área E Perímetro No Qgis | 30 |
| 5.3 | Explicação Do Código: Criação Automática De Um Mapa Padrão Em Paisagem E Retrato. | 31 |
| 5.3.1 | Explicação Do Código Linha A Linha: | 32 |
| 5.3.2 | Intrução Para Uso Do Código E Bom Funcionamento:..... | 33 |
| 5.3.3 | Código Completo: Criação Automática De Uma Mapa Pré-Definida (Anexo 3) (Figuras 15), (Figuras 16), (Figuras 17), (Figuras 18), (Figuras 19). | 34 |
| 5.3.4 | Passo A Passo Para O Uso Correto Do Código..... | 37 |
| 5.3.5 | Importância Do Código Sobre A Automação De Mapas..... | 38 |
| 5.3.6 | Teste Para Demonstração Do Bom Funcionamento Do Código De Automação De Mapas | 38 |

| | | |
|-------|--|----|
| 5.4 | Automatização Da Análise Espacial No Qgis: Desenvolvimento De Um Código Para Avaliação Temporal De Infraestruturas Urbanas E Rurais No Sea Geo | 41 |
| 5.4.1 | Apresentação Dos Tópicos Avaliados No Sea Geo..... | 42 |
| 5.4.2 | Critérios Para Atribuição De Notas No Sea Geo..... | 43 |
| 5.4.3 | Estrutura Do Código Em Python..... | 44 |
| 5.4.4 | Como Utilizar O Código No Qgis | 50 |
| 5.4.5 | Código Para Automação Do Mapa Do Sea Geo Completo (Anexo 4) (Figura 34). (Figura 35). (Figura 36). (Figura 37). | 51 |
| 5.4.6 | Teste Do Funcionamento Completo Do Código De Automação Do Sea Geo... .. | 53 |
| 5.4.7 | Modelo De Relatório Para Complementação Do Mapa Sea Geo..... | 57 |
| 5.4.8 | Modelo Completo Do Relatório (Anexo 5) | 61 |
| 5.5 | Modelos Prontos De Layout Para Todos Os Mapas (Anexo 6): | 61 |
| 6 | A IMPORTÂNCIA DO SEA GEO AMAZÔNICO PARA ANÁLISES TEMPORAIS E SOCIOAMBIENTAIS..... | 62 |
| 6.1 | Monitoramento Das Transformações Socioeconômicas..... | 62 |
| 6.2 | Avaliação De Impactos Ambientais E Sustentabilidade..... | 63 |
| 6.3 | Suporte À Formulação De Políticas Públicas..... | 64 |
| 6.4 | Aplicação Em Pesquisas Acadêmicas E Científicas..... | 64 |
| 6.5 | Aplicação Em Pesquisas Acadêmicas E Científicas: Estudo De Caso Em Quintais Produtivos Na Ilha De Caratateua, Belém/Pa..... | 65 |
| 6.6 | Aplicação Em Pesquisas Acadêmicas E Científicas: Análise Regressiva De Sistemas Agrocológicos: Um Estudo De Caso No Assentamento Paulo Fonteles, Distrito De Mosqueiro, Belém- Pa..... | 68 |
| 6.7 | Metodologia Do Sea Geo Amazônico | 70 |
| 7 | CONCLUSÕES | 72 |
| 8 | REFERÊNCIAS BIBLIOGRÁFICAS | 74 |
| | ANEXO 1 | 76 |
| | ANEXO 2 | 76 |
| | ANEXO 3 | 76 |
| | ANEXO 4..... | 76 |
| | ANEXO 5 | 76 |
| | ANEXO 6..... | 76 |

1 INTRODUÇÃO

O uso de Sistemas de Informações Geográficas (SIG) tem se expandido significativamente, especialmente nas áreas de geoprocessamento e análise espacial. No entanto, muitos processos dentro desses sistemas ainda dependem de tarefas manuais que podem ser lentas e propensas a erros. O avanço da tecnologia, aliado ao desenvolvimento de ferramentas de automação, tem permitido a otimização de diversos fluxos de trabalho em SIG, resultando em maior eficiência e precisão na realização de análises complexas. A crescente necessidade de maior agilidade e confiabilidade nos resultados de análises geoespaciais é uma questão crucial em muitos setores, como planejamento urbano, monitoramento ambiental e gestão territorial.

Neste contexto, o presente trabalho tem como objetivo principal a criação e implementação de ferramentas automatizadas, desenvolvidas em Python, para a realização de tarefas específicas no QGIS, um dos softwares de SIG mais utilizados atualmente. A automação proposta visa resolver problemas recorrentes no geoprocessamento, como a conversão de sistemas de coordenadas e a execução de cálculos geoespaciais, como áreas e perímetros, de forma eficiente e sem erros humanos. Adicionalmente, as ferramentas a serem desenvolvidas permitirão a personalização automática de layouts cartográficos, otimizando a criação de mapas temáticos de acordo com os requisitos do usuário.

A abordagem adotada neste trabalho é fundamentada na utilização da linguagem Python, que possui bibliotecas amplamente suportadas pelo QGIS, como o PyQGIS. Estas bibliotecas permitem a manipulação de camadas geoespaciais e a implementação de processos de automação, oferecendo uma solução prática para a melhoria da produtividade dos profissionais da área. O processo de automação será centrado na conversão de geometria para o sistema de coordenadas SIRGAS 2000, especificamente para zonas UTM, e na execução de cálculos precisos de áreas e perímetros, elementos essenciais em diversos tipos de análises geoespaciais.

Além disso, o trabalho também aborda a necessidade de geração de relatórios e layouts de mapas personalizados. A criação de um layout automatizado para mapas temáticos será um dos pontos principais do estudo, com o objetivo de facilitar a produção de documentos cartográficos padronizados e de fácil leitura, sem a necessidade de intervenções manuais. A integração entre a automatização de tarefas de processamento geoespacial e a criação de layouts temáticos oferece uma solução prática e eficiente para usuários que

necessitam de resultados rápidos e precisos, principalmente em contextos em que há a necessidade de realizar diversas análises em grandes volumes de dados.

A relevância deste estudo está no impacto que as ferramentas automatizadas têm no processo de tomada de decisão em diversas áreas, como planejamento urbano, gestão de recursos naturais e monitoramento ambiental. A automação contribui para a redução do tempo gasto na execução de tarefas repetitivas, melhora a consistência dos resultados e permite aos profissionais focar em aspectos mais analíticos e interpretativos do trabalho. Ao desenvolver ferramentas que automatizam processos como a conversão de coordenadas e a criação de layouts de mapas, este trabalho visa não apenas otimizar o trabalho de profissionais da área, mas também ampliar as possibilidades de análise e apresentação de resultados em projetos geoespaciais.

1.1 Objetivos

Este estudo tem como objetivo principal o desenvolvimento de uma metodologia de automação que, ao ser implementada, possibilite a conversão precisa de dados geoespaciais para o sistema de coordenadas SIRGAS 2000/UTM, a execução de cálculos geoespaciais e a criação automatizada de layouts cartográficos. Com base em bibliotecas Python e PyQGIS, busca-se oferecer uma solução prática e eficiente para a realização de análises e a criação de produtos cartográficos. Ao mesmo tempo, este trabalho busca demonstrar a eficácia da automação na melhoria da qualidade e da produtividade no uso de SIGs.

1.2 Estrutura Do Trabalho

O presente trabalho foi desenvolvido de maneira estruturada para atender aos objetivos estabelecidos, dividindo-se em etapas claras que abordam a automação de tarefas no QGIS com a utilização da linguagem Python. O desenvolvimento foi conduzido com foco em três grandes frentes: a criação de ferramentas automatizadas para o processamento de dados geoespaciais, a implementação de cálculos de áreas e perímetros, e a construção de um layout cartográfico automatizado. Cada uma dessas etapas foi tratada com uma abordagem metodológica própria, sempre focada na otimização de processos dentro do QGIS.

A principal abordagem adotada foi a criação de scripts em Python para automatizar tarefas repetitivas e complexas, como a conversão de sistemas de coordenadas para o sistema SIRGAS 2000/UTM, o cálculo de áreas e perímetros de formas geométricas e a geração

automática de layouts cartográficos. Essas automações foram desenvolvidas com o objetivo de reduzir o tempo necessário para realizar essas operações, aumentar a precisão dos cálculos e facilitar a geração de mapas temáticos de forma padronizada.

O código desenvolvido ao longo deste trabalho é flexível e adaptável, permitindo que o usuário selecione as camadas de interesse e realize as operações de forma simples e eficiente. A escolha da linguagem Python, juntamente com a utilização da biblioteca PyQGIS, oferece uma base robusta e escalável para a automação no ambiente QGIS, além de possibilitar a integração com outras ferramentas de análise geoespacial.

A metodologia adotada também incluiu a criação de tutoriais detalhados para que os usuários possam aplicar as soluções desenvolvidas de forma prática e sem dificuldades, garantindo que os resultados do trabalho possam ser replicados em diferentes contextos de geoprocessamento. Dessa forma, este trabalho não só contribui para o avanço da automação no QGIS, mas também oferece uma ferramenta útil e acessível para profissionais da área.

O objetivo final do estudo foi desenvolver uma solução completa que integre essas ferramentas de automação e facilite o trabalho de profissionais de geoprocessamento, tornando processos complexos e demorados mais rápidos e precisos.

2 CONFIGURAÇÃO E AMBIENTE DE DESENVOLVIMENTO NO QGIS

Este trabalho foi desenvolvido utilizando o **QGIS versão 3.34.14-Prizren**, uma das versões mais recentes e estáveis do software no momento da execução. O QGIS, software de código aberto amplamente utilizado para análise espacial e geoprocessamento, oferece um ambiente robusto para o desenvolvimento de ferramentas automatizadas, como o que foi proposto neste trabalho. A versão do QGIS utilizada, identificada pelo código de versão **0cdaf6d9**, foi escolhida por sua estabilidade e suporte a diversas ferramentas de automação e processamento geoespacial.

A **versão Qt** instalada é a **5.15.13**, que é uma das versões mais recentes e oferece suporte para a criação de interfaces gráficas robustas, o que possibilitou o desenvolvimento de scripts Python no QGIS com interfaces interativas e eficientes. O **Python 3.12.8**, versão atual da linguagem no momento, foi utilizado para a implementação do código de automação, garantindo a compatibilidade com as bibliotecas e funcionalidades mais recentes do QGIS.

O trabalho também utilizou várias bibliotecas geoespaciais que fazem parte da infraestrutura do QGIS, como o **GDAL/OGR 3.9.3**, utilizado para o processamento de dados raster e vetoriais, e o **PROJ 9.5.0**, que fornece suporte a transformações de coordenadas, essencial para a conversão de sistemas de referência espacial no trabalho. O **GEOS 3.13.0-CAPI-1.19.0** foi empregado para operações de geometria e análise espacial, enquanto a **base de dados EPSG v11.016** foi utilizada para garantir a precisão na definição de sistemas de coordenadas e projeções cartográficas.

A versão **SQLite 3.46.1** foi empregada no gerenciamento de banco de dados local, permitindo a manipulação eficiente de dados espaciais. Além disso, foi utilizado o **PostgreSQL versão 16.2**, com o plugin **PostGIS**, para o armazenamento e análise de grandes volumes de dados espaciais, essenciais para a realização de operações complexas no QGIS. O **Spatialite 5.1.0** e o **QWT 6.3.0** também estavam ativos, sendo empregados no processamento de dados espaciais e na criação de interfaces gráficas.

Os plugins Python ativos durante o trabalho incluíram o **quick_map_services 0.19.37**, que facilita a integração de serviços de mapas base com o QGIS, e o **db_manager 0.1.20**, utilizado para o gerenciamento de bancos de dados espaciais dentro do QGIS. O **grassprovider 2.12.99** foi integrado para acessar e utilizar as funcionalidades do GRASS GIS diretamente no QGIS, e o **MetaSearch 0.3.6** foi empregado para a busca e integração de dados espaciais. O plugin **processing 2.12.99** foi essencial para a execução de ferramentas de geoprocessamento, como a análise de geometria e a geração de mapas temáticos.

Esse conjunto de ferramentas e versões garantiu que o trabalho fosse realizado de forma eficiente, integrando recursos poderosos de análise e automação, fundamentais para a execução das operações propostas no TCC.

3 CONFIGURAÇÃO DO SISTEMA E HARDWARE UTILIZADO

O trabalho foi desenvolvido em um computador com configuração robusta, adequado para o processamento e manipulação de grandes volumes de dados geoespaciais. A máquina utiliza um processador Intel Core i7 de 3.40 GHz, com 12 GB de RAM, proporcionando bom desempenho para o processamento de dados no QGIS e execução de scripts Python. O sistema operacional utilizado é o Windows 10, versão 22H2, garantindo a estabilidade e compatibilidade com as ferramentas utilizadas no desenvolvimento do projeto. Essa configuração foi suficiente para realizar todas as etapas do trabalho de forma eficiente, desde a análise de dados até a implementação das ferramentas de automação.

4 DIRETRIZES DE IMPLEMENTAÇÃO E DESENVOLVIMENTO DE CÓDIGOS PADRÃO

O uso de diretrizes padronizadas no desenvolvimento de scripts e programas é fundamental, especialmente em projetos que envolvem automação e geoprocessamento, como os realizados no QGIS. Essas diretrizes visam proporcionar uma estrutura clara e coesa, permitindo que qualquer código escrito seja facilmente compreendido, modificado e utilizado por diferentes membros da equipe ou usuários futuros.

Uma das principais vantagens de adotar diretrizes de implementação é a melhoria da colaboração entre os membros da equipe. Quando todos seguem uma mesma estrutura de codificação, torna-se muito mais fácil compartilhar e integrar diferentes partes do trabalho. A padronização também facilita a integração do código com outras ferramentas e sistemas, como bancos de dados ou APIs externas, proporcionando maior flexibilidade e escalabilidade ao sistema como um todo.

Além disso, o desenvolvimento de códigos conforme diretrizes estabelecidas contribui para a eficiência do processo de codificação. Com padrões claros e bem definidos, evita-se retrabalho e a necessidade de revisões constantes, uma vez que o código já está alinhado com as melhores práticas desde o início. As diretrizes também promovem a redução de erros, já que o uso de convenções torna o código mais legível, organizado e fácil de depurar.

Por fim, a implementação de códigos padronizados facilita a documentação e o treinamento de novos desenvolvedores. Códigos bem estruturados e seguidos por convenções específicas tornam-se mais acessíveis para quem está aprendendo a trabalhar com a tecnologia utilizada, além de permitir uma transição mais tranquila entre diferentes versões ou equipes.

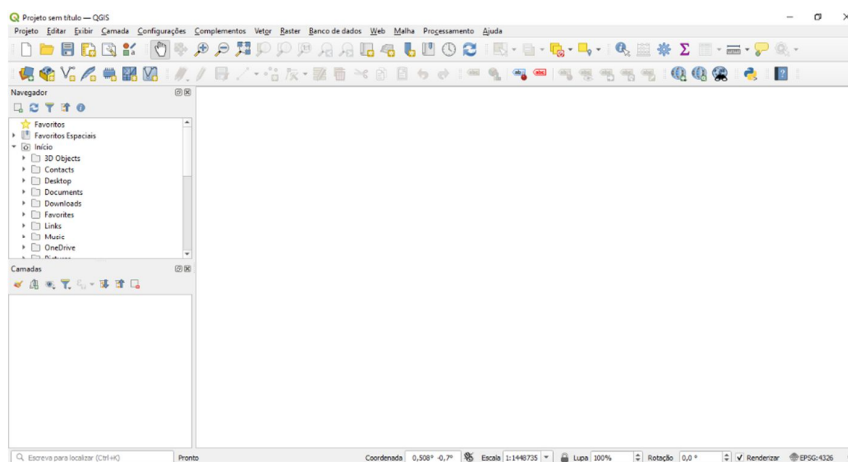
Portanto, estabelecer um conjunto de diretrizes de implementação e desenvolvimento de códigos padrão é um passo essencial para garantir a qualidade, a sustentabilidade e o sucesso dos projetos, permitindo que o código evolua de maneira controlada e eficiente ao longo do tempo.

4.1 Passos Fundamentais Para Garantir A Execução Adequada Dos Códigos

Para rodar um código Python no QGIS, siga as etapas descritas abaixo:

- **Abrir o QGIS:** O primeiro passo é iniciar o QGIS em seu computador. Certifique-se de que a versão instalada seja compatível com os códigos que você deseja executar (Figura 1).

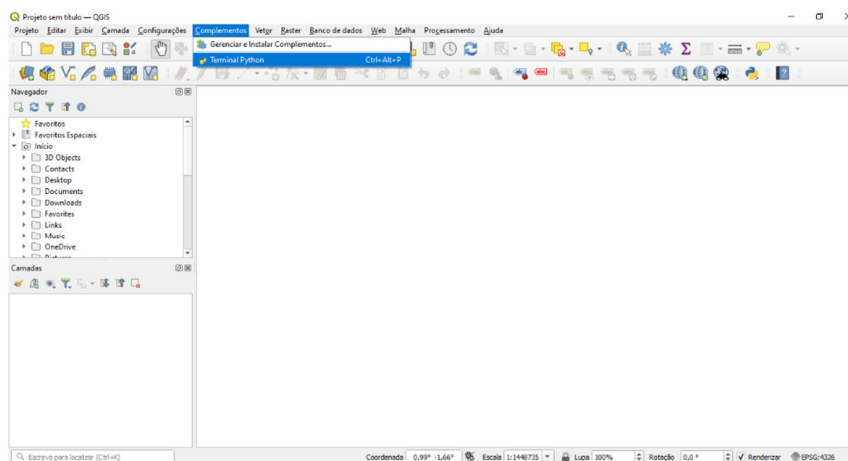
Figura 1– Página Inicial do QGIS



Fonte: Desenvolvido pelo Autor.

- **Acessar o Terminal Python:** No canto superior do QGIS, vá até o menu **Complementos** e clique em **Terminal Python**. Como atalho, você também pode usar o comando **Ctrl + Alt + P** para abrir diretamente o terminal (Figura 2).

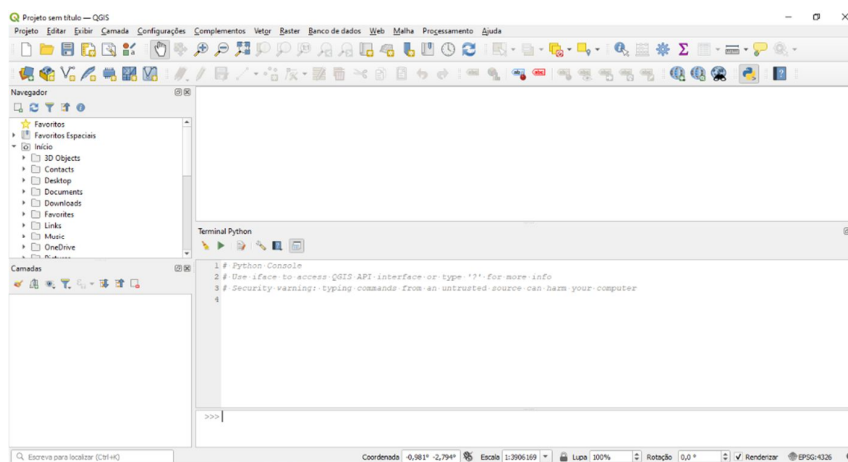
Figura 2-Demonstração de Localização do Terminal Python.



Fonte: Desenvolvido pelo Autor.

- **Abrir o Terminal Python:** Ao fazer isso, o terminal Python será aberto na parte inferior da tela do QGIS. Este espaço é onde você poderá digitar ou colar o código que deseja executar (Figura 3).

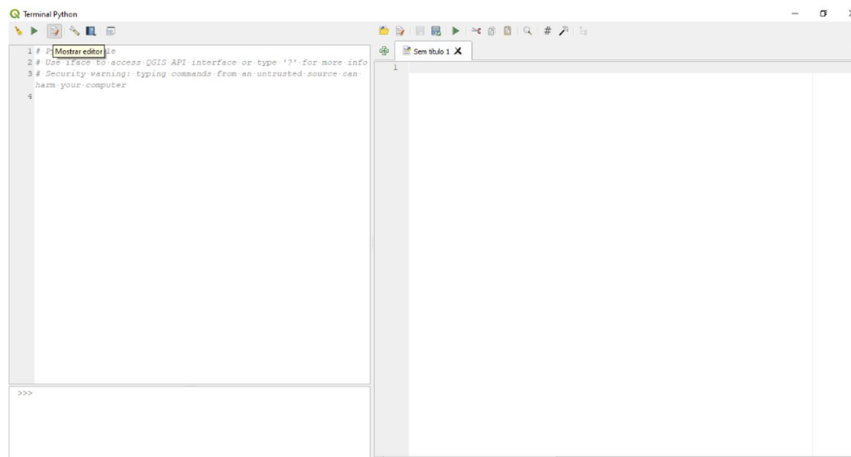
Figura 3 - Demonstração do Terminal Python Aberto.



Fonte: Desenvolvido pelo Autor.

- **Mostrar o Editor de Scripts:** Se preferir editar o código antes de rodá-lo, clique em **Mostrar Editor**, que exibirá uma interface mais confortável para realizar modificações no script. Aqui, você pode copiar o código e fazer ajustes conforme necessário (Figura 4).

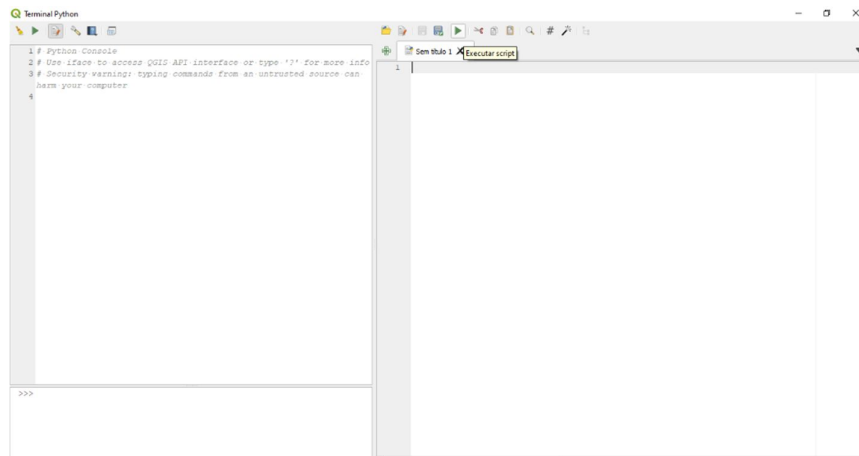
Figura 4 - Demonstração do Terminal Python com a Função Mostrar Editor Aberta.



Fonte: Desenvolvido pelo Autor.

- **Executar o Código:** Após verificar e ajustar o código conforme necessário, clique no botão verde de **Executar Script** para rodar o código no QGIS. O código será executado, e os resultados aparecerão no painel de resultados ou nas camadas do mapa, dependendo da função do código (Figura 5).

Figura 5 - Demonstração do Botão de Execução do Script.



Fonte: Desenvolvido pelo Autor.

5 INSTRUÇÕES E EXEMPLOS DE CÓDIGOS PARA AUTOMAÇÃO DE PROCESSOS NO QGIS: GUIA COMPLETO PARA IMPLEMENTAÇÃO E EXECUÇÃO

5.1 Explicação Do Código: Identificador De Zona Utm De Um Shapefile No Qgis

- **Instruções Adicionais:**

- **Adicionar o Shapefile de Interesse:** Certifique-se de que o shapefile desejado já foi carregado no QGIS antes de rodar o código. Se você não tiver o shapefile carregado, será necessário importá-lo. Caso o shapefile tenha muitas feições, pode ser necessário selecionar a área de interesse e exportar uma nova camada para uma análise mais focada.
- **Selecionar Feições Relevantes:** Caso o shapefile tenha muitas feições (por exemplo, várias áreas geográficas), é recomendável realizar uma seleção das feições que você deseja analisar, utilizando as ferramentas de seleção do QGIS. Após selecionar, você pode exportar essas feições para uma nova camada (arquivo shapefile ou camada temporária) para facilitar o cálculo da zona UTM.

- **Importância do Código:**

Este código é de extrema importância para projetos de geoprocessamento que envolvem análise espacial de dados geográficos em um Sistema de Coordenadas de Referência Universal, como o UTM. Ele automatiza o processo de identificação da zona UTM, que é um passo essencial para realizar transformações de coordenadas e cálculos precisos em sistemas de coordenadas projetadas. Além disso, ele oferece a flexibilidade de identificar automaticamente o hemisfério, permitindo uma análise mais precisa e eficiente. Com isso, os usuários do QGIS podem economizar tempo e reduzir erros na preparação de dados geoespaciais.

- **Importação da Biblioteca Qgis**

Estamos importando a biblioteca `qgis.core`, que fornece a interface para interagir com o QGIS programaticamente. No caso, estamos utilizando `QgsProject`, que permite acessar o projeto QGIS ativo e realizar operações com as camadas carregadas no projeto.

- **Definição da Função: `calcular_zona_utm()`**

Definir a função que calculará a zona UTM e o hemisfério para uma camada selecionada no QGIS. A função recebe como argumento `cam_layer`, que representa a camada ativa ou a camada de interesse do shapefile carregado.

- **Obtenção da Extensão da Camada:**

A função `extent()` retorna as coordenadas mínimas e máximas (X, Y) da camada selecionada, representando sua área de abrangência no espaço geográfico. Essas coordenadas são essenciais para calcular a localização central da camada.

- **Cálculo das Coordenadas Médias (Longitude e Latitude)**

As coordenadas mínimas e máximas são somadas e divididas por 2 para obter as coordenadas médias de longitude e latitude. Isso ajuda a identificar o centro geográfico da camada e fornece os dados necessários para determinar a zona UTM.

- **Cálculo da Zona UTM**

O cálculo da zona UTM é feito com base na longitude média. A Terra é dividida em 60 zonas longitudinais de 6 graus de largura, numeradas de 1 a 60. Este código determina em qual zona a camada está, com base na fórmula $(\text{longitude} + 180) / 6$, onde a longitude é ajustada para o intervalo correto de zonas UTM.

- **Determinação do Hemisfério**

Com base na latitude média, o código determina se a camada está no Hemisfério Norte ou Sul. Se a latitude for maior ou igual a zero, o hemisfério é o Norte, caso contrário, o hemisfério é o Sul.

- **Retorno da Zona e Hemisfério**

A função retorna os valores calculados para a zona UTM e o hemisfério. Estes valores podem ser utilizados em outras partes do código ou exibidos ao usuário.

- **Acesso à Camada Ativa no QGIS**

`iface.activeLayer()` obtém a camada ativa no QGIS, ou seja, a camada que está selecionada no momento da execução do código. A variável `camada_selecionada` armazena essa camada para posterior análise.

- **Verificação e Execução do Cálculo**

O código verifica se uma camada foi selecionada no QGIS. Caso haja uma camada ativa, ele chama a função `calcular_zona_utm()` e exibe o resultado no console, mostrando a zona UTM e o hemisfério. Caso contrário, ele exibe uma mensagem de erro, indicando que nenhuma camada foi selecionada.

5.1.1 Código: Identificação De Zona Utm De Um Shapefile No Qgis (Anexo 1) (Figura 6).

Figura 6 – Código Completo de Identificação de Zona UTM de um Shapefile no Qgis.

```

1 from qgis.core import QgsProject
2
3 def calcular_zona_utm(cam_layer):
4     """
5     Calcula a zona UTM de uma camada selecionada e verifica se é Norte ou Sul.
6     param cam_layer: Camada ativa selecionada.
7     return: Zona UTM e hemisfério (Norte ou Sul).
8     """
9     # Obtém a extensão da camada
10    extent = cam_layer.extent()
11    # Calcula a longitude e latitude médias da extensão da camada
12    longitude = (extent.xMinimum() + extent.xMaximum()) / 2
13    latitude = (extent.yMinimum() + extent.yMaximum()) / 2
14
15    # Calcula a zona UTM
16    zona = int((longitude + 180) / 6) + 1
17
18    # Determina o hemisfério
19    hemisferio = "Norte" if latitude >= 0 else "Sul"
20
21    return zona, hemisferio
22
23 # Obtém a camada ativa no QGIS
24 camada_selecionada = iface.activeLayer() # Usa iface para capturar a camada ativa
25
26 if camada_selecionada:
27     zona_utm, hemisferio = calcular_zona_utm(camada_selecionada)
28     print(f"A zona UTM da camada '{camada_selecionada.name()}' é: (zona_utm) {hemisferio}")
29 else:
30     print("Nenhuma camada está selecionada no momento.")

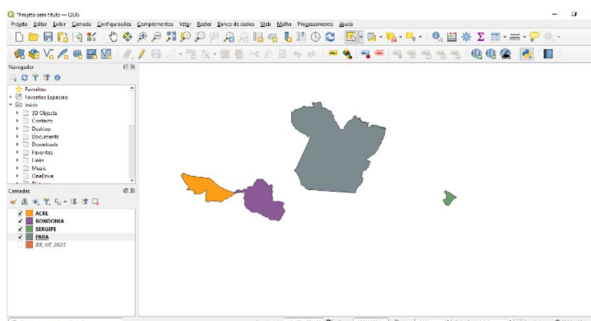
```

Fonte: Desenvolvido pelo Autor.

5.1.2 Testes Do Código: Identificação De Zona Utm De Um Shapefile No Qgis

O código foi testado utilizando dados de municípios de diversos estados brasileiros como parte do processo de validação de sua funcionalidade. Para garantir que o cálculo da zona UTM e a determinação do hemisfério fossem executados corretamente, foram selecionados municípios dos estados do Acre, Rondônia, Pará e Sergipe. Esses municípios foram escolhidos por sua diversidade geográfica, abrangendo áreas tanto no Hemisfério Norte quanto no Hemisfério Sul, e diferentes zonas UTM (Figura 7).

Figura 7 - Demonstração dos Estados Usados para o Teste.



Fonte: Desenvolvido pelo Autor.

Durante os testes, o código demonstrou precisão ao calcular corretamente a zona UTM e o hemisfério de cada município, validando a eficácia e a aplicabilidade do script para análise geoespacial em diferentes contextos regionais (Figura 8).

Figura 8 - Resultados do Funcionamento do Código de Identificação das Zonas UTM.

```

4 >>> exec(Path('C:/Users/Meu Computador/Documents/João Guilherme S
RS/MEU TCC/CÓDIGOS/IDENTIFICADOR_DE_ZONA_UTM.py')).read_text()
5 A zona UTM da camada 'ACRE' é: 19 Sul
6 >>> exec(Path('C:/Users/Meu Computador/Documents/João Guilherme S
RS/MEU TCC/CÓDIGOS/IDENTIFICADOR_DE_ZONA_UTM.py')).read_text()
7 A zona UTM da camada 'RONDONIA' é: 20 Sul
8 >>> exec(Path('C:/Users/Meu Computador/Documents/João Guilherme S
RS/MEU TCC/CÓDIGOS/IDENTIFICADOR_DE_ZONA_UTM.py')).read_text()
9 A zona UTM da camada 'SERGIPE' é: 24 Sul
10 >>> exec(Path('C:/Users/Meu Computador/Documents/João Guilherme S
RS/MEU TCC/CÓDIGOS/IDENTIFICADOR_DE_ZONA_UTM.py')).read_text()
11 A zona UTM da camada 'PARA' é: 22 Sul

```

Fonte: Desenvolvido pelo Autor.

5.2 Explicação Do Código: Conversão De Camada Para Utm Em Sirgas 2000 E Cálculo De Área E Perímetro No Qgis

- **Instruções Adicionais: Adicionar o Shapefile de Interesse e Selecionar Feições Relevantes:**

Antes de executar este código, é essencial que o usuário se certifique de que o shapefile de interesse já está carregado no QGIS. Para isso, basta utilizar o menu Camadas do software e adicionar a camada correspondente. Após carregar o shapefile, é importante selecionar a camada desejada como a camada ativa, garantindo que ela será usada para os cálculos e conversões especificados no script. No caso de shapefiles que possuem muitas feições, como grandes áreas urbanas ou regiões extensas, recomenda-se a seleção de feições específicas que sejam relevantes para a análise. Isso pode ser feito utilizando as ferramentas de seleção do QGIS, como seleção por atributos ou por localização. Além disso, para análises mais focadas e precisas, as feições selecionadas podem ser exportadas para uma nova camada temporária ou shapefile. Esse cuidado garante que os resultados gerados pelo código correspondam aos interesses específicos do usuário, otimizando o desempenho e a eficiência das operações.

- **Importância do Código:**

Este código é uma ferramenta poderosa para usuários de geoprocessamento que trabalham com análises espaciais precisas em diferentes zonas UTM. Ele automatiza o processo de conversão de camadas para o sistema de referência de coordenadas SIRGAS 2000 UTM, garantindo que os dados estejam alinhados com os padrões cartográficos e prontos para integração com outros sistemas ou camadas. A possibilidade de calcular áreas e perímetros de forma automatizada, com unidades como metros quadrados e hectares, facilita a obtenção de informações detalhadas sobre feições geográficas, o que é crucial em aplicações como planejamento urbano, gestão ambiental e agricultura. Além disso, a automação reduz o tempo e o esforço necessários para realizar essas tarefas manualmente, minimizando erros e permitindo que os profissionais se concentrem em análises mais estratégicas. A implementação desse código no QGIS demonstra como a programação pode ser usada para melhorar a eficiência e a qualidade do trabalho em geoprocessamento.

- **Importação de Bibliotecas Necessárias:**

Este trecho importa as bibliotecas essenciais para o funcionamento do código. `QgsProject` e `iface` permitem acessar o projeto e a camada ativa no QGIS, enquanto `QgsCoordinateReferenceSystem` e `QgsCoordinateTransform` são usados para realizar as transformações de sistemas de coordenadas.

- **Definição da Função `converter_e_calcular`:**

A função `converter_e_calcular` recebe como parâmetro a camada ativa no QGIS (`cam_layer`) e encapsula todo o processo de conversão e cálculo. Essa abordagem modular facilita o reuso do código em diferentes contextos.

- **Verificação da Validade da Camada:**

Este trecho verifica se uma camada foi selecionada. Caso contrário, exibe uma mensagem informando ao usuário que nenhuma camada está ativa e interrompe a execução para evitar erros.

- **Cálculo da Zona UTM e Hemisfério:**

O código calcula a zona UTM com base na longitude média da camada e determina o hemisfério (Norte ou Sul) com base na latitude. Isso é essencial para definir o EPSG correto para a camada em SIRGAS 2000 UTM.

- **Transformação da Geometria:**

Aqui, o sistema de referência da camada é convertido para o SIRGAS 2000 UTM adequado. As geometrias são transformadas e armazenadas para cálculos posteriores.

- **Cálculos de Área e Perímetro:**

Este trecho calcula a área e o perímetro de cada feição da camada, somando os valores para obter o total. As unidades de área (m², ha) e perímetro (m, ha) são derivadas diretamente desses cálculos.

- **Exibição dos Resultados:**

Os resultados são exibidos no console Python do QGIS, incluindo o nome da camada, a zona UTM e os valores de área e perímetro calculados.

5.2.1 Código: Conversão De Camada Para Utm Em Sirgas 2000 E Cálculo De Área E Perímetro No Qgis (Anexo 2) (Figura 9) (Figura 10).

Figura 9 - Demonstração Completa de Cálculo de Área e Perímetro – Parte 1

```

1 from qgis.core import (
2     QgsProject,
3     QgsCoordinateReferenceSystem,
4     QgsCoordinateTransform,
5     QgsGeometry
6 )
7 from qgis.utils import iface
8
9 def converter_e_calcular(cam_layer):
10     """
11     Converte a camada para o sistema SIRGAS 2000 UTM correto e realiza cálculos de área e perímetro.
12     Exibe os resultados no painel.
13     @param cam_layer: Camada ativa selecionada.
14     """
15     try:
16         # Verificar se a camada está válida
17         if not cam_layer:
18             raise ValueError("Nenhuma camada selecionada! Certifique-se de selecionar a camada desejada antes de executar o código.")
19         return
20
21         # Obter a extensão média (centroide) da camada para determinar a zona UTM
22         extent = cam_layer.extent()
23         longitude = (extent.minimum() + extent.maximum()) / 2
24         latitude = (extent.minimum() + extent.maximum()) / 2
25
26         # Determinar a zona UTM
27         zona = int((longitude + 180) / 6) + 1
28         hemisferio = "Sul" if latitude < 0 else "Norte"
29
30         # Definir o código EPSG correto para SIRGAS 2000 UTM
31         if hemisferio == "Sul":
32             epsg_code = 31971 + (zona - 17)  # EPSG para zonas UTM no Hemisfério Sul
33         else:
34             epsg_code = 31965 + (zona - 17)  # EPSG para zonas UTM no Hemisfério Norte
35
36         # Criar o sistema de referência CRS SIRGAS 2000 UTM
37         crs_destino = QgsCoordinateReferenceSystem(f"EPSG:{epsg_code}")
38
39         # Transformar a geometria da camada
40         transform = QgsCoordinateTransform(
41             cam_layer.crs(), crs_destino, QgsProject.instance()
42         )
43
44         nova_geometria = []
45         for feat in cam_layer.getFeatures():
46             geom = feat.geometry()
47             geom.transform(transform)
48             nova_geometria.append(geom)

```

Fonte: Desenvolvido pelo Autor.

Figura 10 - Demonstração Completa de Cálculo de Área e Perímetro – Parte 2

```

48
49 # Calcular os valores de área e perímetro
50 area_total_m2:=0
51 perimetro_total_m:=0
52
53 for geom in nova_geometria:
54     area_total_m2+=geom.area()
55     perimetro_total_m+=geom.length()
56
57 area_total_ha:=area_total_m2/.10000 # Conversão de m² para hectares
58 perimetro_total_ha:=perimetro_total_m/.10000 # Conversão de metros para hectares
59
60 # Exibir os resultados no painel
61 print(f"Nome da camada utilizada: {(cam_layer.name())}")
62 print(f"Zona UTM convertida: {zona} -- Hemisfério: {hemisferio}")
63 print(f"Área total: {(area_total_m2:.2f)} m²")
64 print(f"Área total: {(area_total_ha:.2f)} ha")
65 print(f"Perímetro total: {(perimetro_total_m:.2f)} m")
66 print(f"Perímetro total: {(perimetro_total_ha:.2f)} ha")
67
68 except Exception as e:
69     print(f"Ocorreu um erro: {e}")
70
71 # Obter a camada ativa no QGIS
72 camada_selecionada:=iface.activeLayer()
73
74 # Executar a função com a camada selecionada
75 converter_e_calcular(camada_selecionada)

```

Fonte: Desenvolvido pelo Autor.

5.2.2 Testes Do Código: Conversão De Camada Para Utm Em Sirgas 2000 E Cálculo De Área E Perímetro No Qgis

O código foi testado utilizando uma camada representando a área do município de Ananindeua, no estado do Pará, como base. Após carregar e selecionar a camada no QGIS, o script foi executado com sucesso, e os resultados apresentados corresponderam aos dados reais da região. A zona UTM foi corretamente identificada como 22S, de acordo com a localização geográfica do município, e os cálculos de área e perímetro foram validados com base em dados oficiais. Esse teste demonstra a precisão e a aplicabilidade do código em análises espaciais, destacando sua utilidade para aplicações práticas em geoprocessamento (Figura 11).

Figura 11 - Resultado do Funcionamento do Código de Cálculo de Área e Perímetro.

```

4 >>> exec(Path('C:/Users/MEUCOM~1/AppData/Local/Temp/tmp9jhb2zww')
      .read_text())
5 Nome da camada utilizada: ANANINDEUA
6 Zona UTM convertida: 22 -- Hemisfério Sul
7 Área total: 190837822.67 m²
8 Área total: 19083.78 ha
9 Perímetro total: 102950.73 m
10 Perímetro total: 10.30 ha

```

Fonte: Desenvolvido pelo Autor.

5.3 Explicação Do Código: Criação Automática De Um Mapa Padrão Em Paisagem E Retrato.

O código desenvolvido é um script em Python para ser executado no console do QGIS. Ele automatiza a criação de layouts de mapas, utilizando layouts pré-configurados e adaptando as extensões de visualização do Mapa Principal e do Mapa de Localização com base em camadas específicas. Este código é ideal para projetos que exigem padronização e eficiência na criação de mapas, sendo particularmente útil para trabalhos acadêmicos, como TCCs, ou no contexto profissional.

- **Objetivo:**
 - Criar um layout de mapa no QGIS usando layouts predefinidos, como MODELO_PADRAO_PAISAGEM ou MODELO_PADRAO_RETRATO.
 - Configurar dois mapas dentro do layout:
 - Mapa Principal: Representa o conteúdo principal
 - Mapa de Localização: Representa um mapa auxiliar de localização.
 - Garantir que a extensão dos mapas seja adaptável ao tamanho da moldura, preservando proporções.
- **Estrutura do Código:**
 - Importação de Bibliotecas:
As bibliotecas do QGIS (qgis.core) e do PyQt5 (PyQt5.QtXml) são usadas para interagir com o projeto QGIS, gerenciar layouts e manipular documentos XML.
- **Funções:**
 - obter_camadas_do_grupo:

- Localiza todas as camadas dentro de um grupo especificado no painel de camadas do QGIS.
- Importante para garantir que os grupos de camadas, como "MAPA PRINCIPAL" e "MAPA DE LOCALIZAÇÃO", sejam utilizados corretamente.
- ajustar_extensao_para_moldura:
 - Calcula uma extensão ajustada para que o conteúdo do Mapa Principal caiba perfeitamente na moldura do mapa, mantendo a proporção entre largura e altura.
- gerar_mapa_com_modelo:
 - Função principal:
 - Carrega um layout preexistente.
 - Configura o Mapa Principal e o Mapa de Localização usando camadas de referência especificadas pelo usuário.
 - Ajusta a extensão do Mapa Principal para caber na moldura do mapa.
 - Adiciona o novo layout ao Gerenciador de Layouts do QGIS.

5.3.1 Explicação Do Código Linha A Linha:

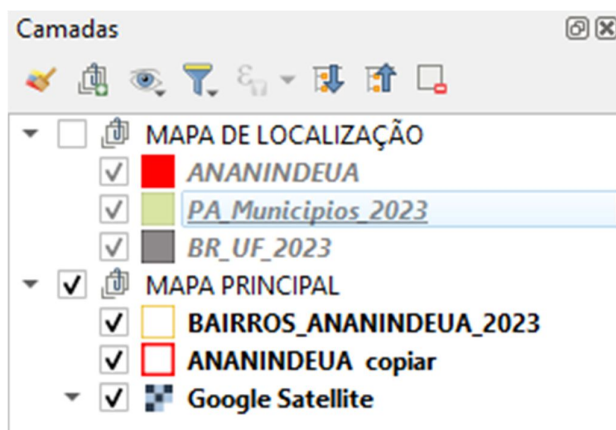
- **Importação das Bibliotecas:**
 - O código utiliza módulos do qgis.core para acessar funcionalidades do QGIS e do PyQt5.QtXml para manipular arquivos XML.
 - os é usado para gerenciar caminhos de arquivos temporários.
- **Função obter_camadas_do_grupo:**
 - Localiza um grupo de camadas pelo nome no painel de camadas do QGIS.
 - Retorna uma lista de camadas contidas no grupo.
- **Função ajustar_extensao_para_moldura:**
 - Calcula uma nova extensão para ajustar o mapa à moldura, mantendo as proporções.
- **Função gerar_mapa_com_modelo:**
 - Passo a Passo:
 - Carregar o Layout Modelo:
 - Carrega um layout existente no Gerenciador de Layouts.

- Salva o layout em um arquivo temporário para manipulá-lo.
- Configurar o Mapa Principal:
 - Localiza a camada de referência definida pelo usuário.
 - Ajusta a extensão da camada para caber na moldura do mapa.
- Configurar Mapa de Localização:
 - Similar ao Mapa Principal, mas sem ajuste de moldura.
- Adicionar o Gerenciador:
 - Adiciona o novo layout ao Gerenciador de Layouts.

5.3.2 Intrução Para Uso Do Código E Bom Funcionamento:

- **Pré-requisitos:**
 - Organizar as Camadas:
 - No painel de Camadas do Qgis, organize as camadas em dois grupos: MAPA PRINCIPAL (Camadas principais) e MAPA DE LOCALIZAÇÃO (Camadas auxiliares) (Figura 12).

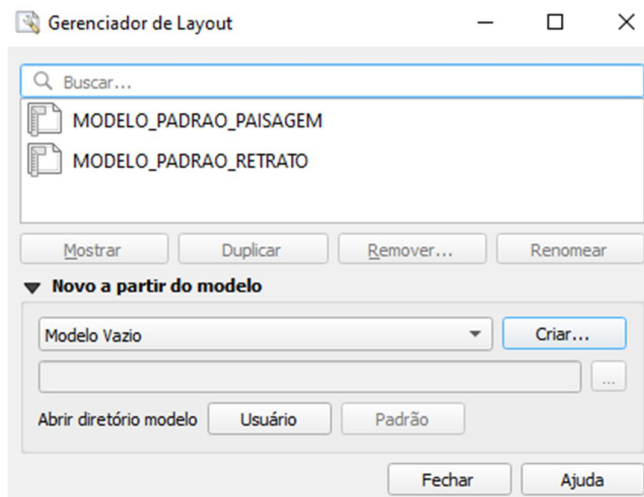
Figura 12 - Organização das Camadas



Fonte: Desenvolvido pelo Autor.

- Criar Layouts Predefinidos:
 - Crie dois layouts no Gerenciador de Layouts do QGIS: MODELO_PADRAO_PAISAGEM e MODELO_PADRAO_RETRATO. (Figura 13).

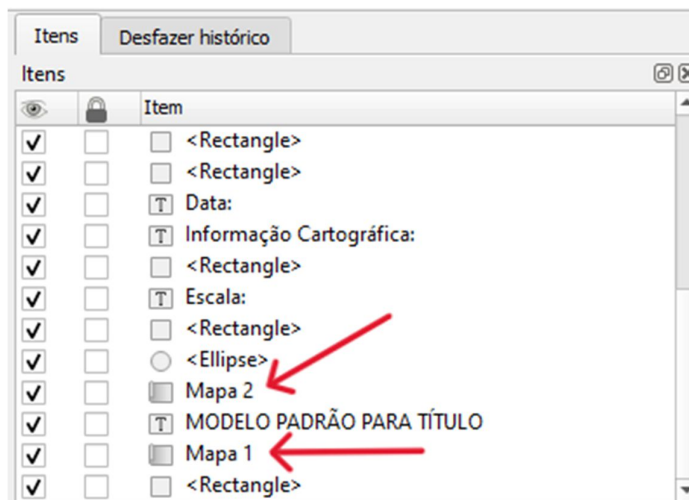
Figura 13 - Condições no Gerenciador de Layout



Fonte: Desenvolvido pelo Autor.

- Certifique-se de que cada layout contenha dois mapas com os IDs: Mapa 1 (para o Mapa Principal) e Mapa 2 (para o Mapa de Localização) (Figura 14).

Figura 14 - Condições no Layout para o Funcionamento do Código



Fonte: Desenvolvido pelo Autor.

5.3.3 Código Completo: Criação Automática De Uma Mapa Pré-Definida (Anexo 3) (Figuras 15), (Figuras 16), (Figuras 17), (Figuras 18), (Figuras 19).

Figura 15 - Código Completo de Automação de um Mapa (Parte 1).

```

1 from qgis.core import (
2     QgsProject,
3     QgsPrintLayout,
4     QgsLayoutItemMap,
5     QgsCoordinateReferenceSystem,
6     QgsReadWriteContext,
7     QgsRectangle,
8 )
9 from PyQt5.QtXml import QDomDocument
10 import os
11
12
13 def ajustar_extensao_para_moldura(extent, mapa):
14     """
15     Ajusta a extensão para caber dentro da moldura do mapa, preservando as proporções.
16
17     Args:
18         extent (QgsRectangle): Extensão original.
19         mapa (QgsLayoutItemMap): Objeto do mapa no layout.
20
21     Returns:
22         QgsRectangle: Extensão ajustada.
23     """
24     # Tamanho da moldura do mapa
25     largura_moldura = mapa.rect().width()
26     altura_moldura = mapa.rect().height()
27
28     # Razão de aspecto da extensão original
29     largura_extent = extent.width()
30     altura_extent = extent.height()
31     razao_aspecto_extent = largura_extent / altura_extent
32
33     # Razão de aspecto da moldura
34     razao_aspecto_moldura = largura_moldura / altura_moldura

```

Fonte: Desenvolvido pelo Autor.

Figura 16 - Código Completo de Automação de um Mapa (Parte 2).

```

36 # Ajustar a extensão com base na razão de aspecto
37 if razao_aspecto_extent > razao_aspecto_moldura:
38     # A extensão é mais larga que a moldura - ajuste a altura
39     nova_altura = largura_extent / razao_aspecto_moldura
40     centro_y = extent.center().y()
41     extent.setMinimum(centro_y - nova_altura / 2)
42     extent.setMaximum(centro_y + nova_altura / 2)
43 else:
44     # A extensão é mais alta que a moldura - ajuste a largura
45     nova_largura = altura_extent * razao_aspecto_moldura
46     centro_x = extent.center().x()
47     extent.setMinimum(centro_x - nova_largura / 2)
48     extent.setMaximum(centro_x + nova_largura / 2)
49
50 return extent
51
52
53 def gerar_mapa_com_modelo(
54     layout_nome,
55     grupo_mapa_principal,
56     grupo_mapa_localizacao,
57     camada_referencia_principal,
58     camada_referencia_localizacao,
59     layout_modelo
60 ):
61     """
62     Gera um mapa com base em um layout existente, configurando dois mapas:
63     - Mapa Principal: Representa o conteúdo principal.
64     - Mapa de Localização: Representa um mapa auxiliar de localização.
65
66     Args:
67         layout_nome (str): Nome do layout a ser criado.
68         grupo_mapa_principal (str): Nome do grupo de camadas para o mapa principal.
69         grupo_mapa_localizacao (str): Nome do grupo de camadas para o mapa de localização.
70         camada_referencia_principal (str): Nome da camada de referência para escala no Mapa Principal.

```

Fonte: Desenvolvido pelo Autor.

Figura 17 - Código Completo de Automação de um Mapa (Parte 3).

```

71 | camada_referencia_localizacao (str): Nome da camada de referência para escala no Mapa de Localização.
72 | layout_modelo (str): Nome do layout modelo para usar (PAISAGEM ou RETRATO).
73 |
74 | try:
75 |     # Obter o projeto atual e gerenciador de layouts
76 |     projeto = QgsProject.instance()
77 |     layout_manager = projeto.layoutManager()
78 |
79 |     # Carregar o layout modelo escolhido
80 |     layout_existente = layout_manager.layoutByName(layout_modelo)
81 |     if not layout_existente:
82 |         raise Exception(f"O layout modelo '{layout_modelo}' não foi encontrado.")
83 |
84 |     # Salvar o layout modelo em um arquivo temporário
85 |     temp_modelo_path = os.path.join(os.path.expanduser("~"), "temp_layout_modelo.qpt")
86 |     context = QgsReadWriteContext()
87 |     if not layout_existente.saveAsTemplate(temp_modelo_path, context):
88 |         raise Exception("Erro ao exportar o modelo do layout existente para arquivo.")
89 |
90 |     # Carregar o modelo do arquivo temporário para um QDomDocument
91 |     dom_document = QDomDocument()
92 |     with open(temp_modelo_path, "r", encoding="utf-8") as file:
93 |         dom_document.setContent(file.read())
94 |
95 |     # Criar um novo layout como QgsPrintLayout e carregar o modelo
96 |     layout = QgsPrintLayout(projeto)
97 |     layout.initializeDefaults()
98 |     if not layout.loadFromTemplate(dom_document, context):
99 |         raise Exception("Erro ao carregar o modelo no novo layout.")
100 |
101 |     # Verificar se o layout já existe
102 |     existing_layouts = [l.name() for l in layout_manager.layouts()]
103 |     if layout_nome in existing_layouts:
104 |         layout_nome += "_novo"
105 |

```

Fonte: Desenvolvido pelo Autor.

Figura 18 - Código Completo de Automação de um Mapa (Parte 4).

```

106 | # Nomear e adicionar o layout ao Gerenciador de Layouts
107 | layout.setName(layout_nome)
108 | layout_manager.addLayout(layout)
109 |
110 | # Configurar o Mapa Principal
111 | mapa1 = layout.itemById("Mapa 1")
112 | if not mapa1 or not isinstance(mapa1, QgsLayoutItemMap):
113 |     raise Exception("O layout modelo não contém um item 'Mapa 1' válido.")
114 |
115 | # Localizar a camada de referência para o Mapa Principal
116 | camada_principal = projeto.mapLayersByName(camada_referencia_principal)
117 | if not camada_principal:
118 |     raise Exception(f"A camada de referência '{camada_referencia_principal}' não foi encontrada.")
119 | extent_principal = camada_principal[0].extent()
120 |
121 | # Ajustar a extensão do Mapa Principal para caber na moldura
122 | extent_ajustado = ajustar_extensao_para_moldura(extent_principal, mapa1)
123 | mapa1.setExtent(extent_ajustado)
124 |
125 | # Configurar o Mapa de Localização
126 | mapa2 = layout.itemById("Mapa 2")
127 | if not mapa2 or not isinstance(mapa2, QgsLayoutItemMap):
128 |     raise Exception("O layout modelo não contém um item 'Mapa 2' válido.")
129 |
130 | # Localizar a camada de referência para o Mapa de Localização
131 | camada_localizacao = projeto.mapLayersByName(camada_referencia_localizacao)
132 | if not camada_localizacao:
133 |     raise Exception(f"A camada de referência '{camada_referencia_localizacao}' não foi encontrada.")
134 | extent_localizacao = camada_localizacao[0].extent()
135 | mapa2.setExtent(extent_localizacao)
136 |
137 | # Informar que o layout foi configurado com sucesso
138 | print(f"O layout '{layout_nome}' foi configurado e está disponível no Gerenciador de Layouts.")
139 |
140 | except Exception as e:

```

Fonte: Desenvolvido pelo Autor.

Figura 19 - Código Completo de Automação de um Mapa (Parte 5).

```

127 == if not mapa2 or not isinstance(mapa2, QgsLayoutItemMap):
128     raise Exception("O layout modelo não contém um item 'Mapa 2' válido.")
129
130 # Localizar a camada de referência para o Mapa de Localização
131 camada_localizacao = projeto.mapLayersByName(camada_referencia_localizacao)
132 == if not camada_localizacao:
133     raise Exception("A camada de referência '(camada_referencia_localizacao)' não foi encontrada.")
134 extent_localizacao = camada_localizacao[0].extent()
135 mapa2.setExtent(extent_localizacao)
136
137 # Informar que o layout foi configurado com sucesso
138 print(f"O layout '{layout_nome}' foi configurado e está disponível no Gerenciador de Layouts.")
139
140 == except Exception as e:
141     print(f"Erro ao gerar o mapa: {e}")
142
143
144 # Configurações do usuário
145 layout_modelo_escolhido = "MODELO_PADRAO_PAISAGEM" # Alternativas: MODELO_PADRAO_RETRATO
146 grupo_mapa_principal = "MAPA_PRINCIPAL" # Nome do grupo principal
147 grupo_mapa_localizacao = "MAPA DE LOCALIZAÇÃO" # Nome do grupo de localização
148 camada_referencia_principal = "Camada_Principal_Ref" # Nome da camada de referência para o Mapa Principal
149 camada_referencia_localizacao = "Camada_Localizacao_Ref" # Nome da camada de referência para o Mapa de Localização
150 nome_layout_novo = "Mapa_Gerado"
151
152 # Executar a função
153 == gerar_mapa_com_modelo
154     nome_layout_novo,
155     grupo_mapa_principal,
156     grupo_mapa_localizacao,
157     camada_referencia_principal,
158     camada_referencia_localizacao,
159     layout_modelo_escolhido
160

```

Fonte: Desenvolvido pelo Autor.

5.3.4 Passo A Passo Para O Uso Correto Do Código

- **Ajustar Configurações no Código:**

Modifique as seguintes variáveis no código para corresponder aos nomes das camadas e grupos de seu projeto (Figura 20):

Figura 20 - Configure esse Trecho para Adaptabilidade para o Funcionamento do Código.

```

144 # Configurações do usuário
145 layout_modelo_escolhido = "MODELO_PADRAO_PAISAGEM" # Alternativas: MODELO_PADRAO_RETRATO
146 grupo_mapa_principal = "MAPA_PRINCIPAL" # Nome do grupo principal
147 grupo_mapa_localizacao = "MAPA DE LOCALIZAÇÃO" # Nome do grupo de localização
148 camada_referencia_principal = "Camada_Principal_Ref" # Nome da camada de referência para o Mapa Principal
149 camada_referencia_localizacao = "Camada_Localizacao_Ref" # Nome da camada de referência para o Mapa de Localização
150 nome_layout_novo = "Mapa_Gerado"

```

Fonte: Desenvolvido pelo Autor.

- **Verificar o Resultado:**

- O novo layout será criado e estará disponível no Gerenciador de Layouts do QGIS.
- Verifique se o Mapa Principal e o Mapa de Localização estão configurados corretamente.

- **Personalização e Exportação:**

- Ajuste manualmente, se necessário, no Gerenciador de Layouts.
- Exporte o layout para PDF ou imagem usando o menu EXPORTAR.

5.3.5 Importância Do Código Sobre A Automação De Mapas

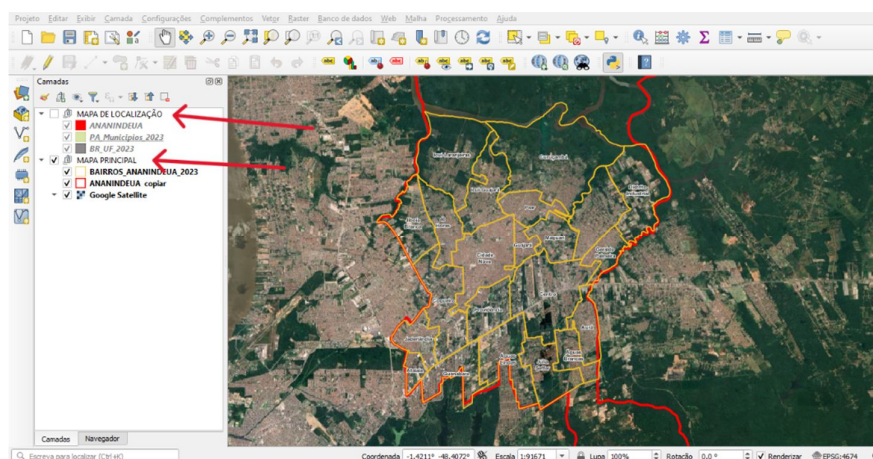
Este script otimiza e automatiza tarefas repetitivas na criação de mapas, oferecendo:

- **Padronização:** Mantém consistência em layouts e mapas.
- **Eficiência:** Reduz o tempo necessário para configurar layouts manualmente.
- **Flexibilidade:** Permite ajustes personalizáveis para diferentes projetos.
- **Escalabilidade:** Pode ser usado em projetos com grande número de mapas.

5.3.6 Teste Para Demonstração Do Bom Funcionamento Do Código De Automação De Mapas

O teste do funcionamento do código foi realizado no ambiente QGIS, onde inicialmente foram criados os grupos necessários no painel de camadas. Foram criados dois grupos principais: o grupo denominado **MAPA PRINCIPAL**, contendo as camadas relacionadas ao conteúdo principal do mapa, como shapefiles de áreas urbanas e limites geográficos, e o grupo **MAPA DE LOCALIZAÇÃO**, que incluía camadas auxiliares, como a visão geral de um estado ou país. As camadas foram verificadas para garantir que estavam corretamente organizadas, visíveis e disponíveis no projeto (Figura 21):

Figura 21 - Demonstração das Camadas e Grupos Criados para o Teste.

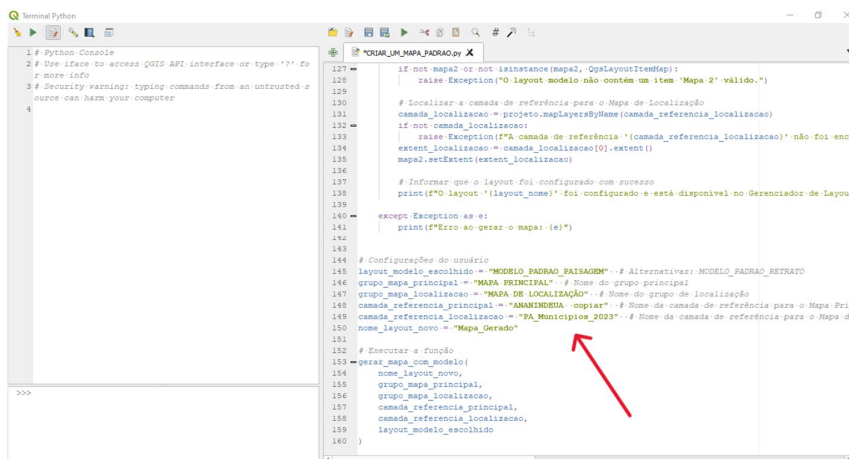


Fonte: Desenvolvido pelo Autor.

Em seguida, o código foi ajustado para identificar os grupos e as camadas de referência específicas. No bloco de configuração do script, as variáveis foram alteradas para

refletir os nomes dos grupos e camadas criados. O layout selecionado foi o **MODELO_PADRAO_PAISAGEM**, o grupo de camadas principais foi definido como **MAPA PRINCIPAL**, e o grupo de camadas auxiliares como **MAPA DE LOCALIZAÇÃO**. Além disso, foi especificada uma camada de referência para cada mapa: **Camada_Urbana** para o Mapa Principal e **Camada_Estado** para o Mapa de Localização. O nome do layout gerado foi definido como **Mapa_Gerado**. (Figura 22).

Figura 22 - Ajuste Realizado para as Camadas em Questão para serem usadas.



```

1 # Python Console
2 # Use Ctrl+Q to access QGIS API interface or type '?' for
3 # more info
4 # Security warning: typing commands from an untrusted s
5 # ource can harm your computer
6
127
128
129
130
131 # Localizar a camada de referência para o Mapa de Localização
132 camada_localizacao = projeto.mapLayersByName(camada_referencia_localizacao)
133 if not camada_localizacao:
134     raise Exception(f"Camada de referência '{camada_referencia_localizacao}' não foi encontrada")
135 extent_localizacao = camada_localizacao[0].extent()
136 mapa2.estender(extent_localizacao)
137
138 # Informar que o layout foi configurado com sucesso
139 print(f"O layout '{layout_nome}' foi configurado e está disponível no Gerenciador de Layouts")
140
141 except Exception as e:
142     print(f"Erro ao gerar o mapa: {e}")
143
144
145 # Configurações do usuário
146 layout_modelo_escolhido = "MODELO_PADRAO_PAISAGEM" # Alternativas: MODELO_PADRAO_RETRATO
147 grupo_mapa_principal = "MAPA PRINCIPAL" # Nome do grupo principal
148 grupo_mapa_localizacao = "MAPA DE LOCALIZAÇÃO" # Nome do grupo de localização
149 camada_referencia_principal = "CAMINHOXUA" # Nome da camada de referência para o Mapa Principal
150 camada_referencia_localizacao = "PA_Municipios_2023" # Nome da camada de referência para o Mapa de Localização
151 nome_layout_novo = "Mapa_Gerado"
152
153 # Executar a função
154 gerar_mapa_com_modelo(
155     nome_layout_novo,
156     grupo_mapa_principal,
157     grupo_mapa_localizacao,
158     camada_referencia_principal,
159     camada_referencia_localizacao,
160     layout_modelo_escolhido
161 )

```

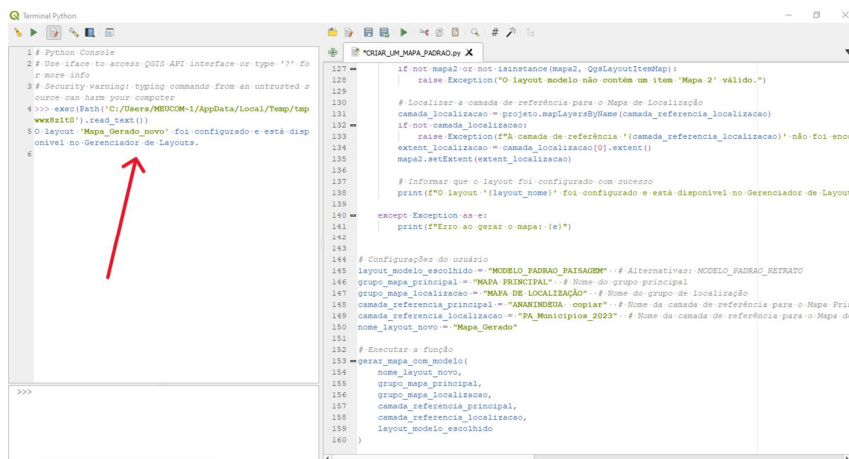
Fonte: Desenvolvido pelo Autor.

Após ajustar as configurações, o código foi executado no console Python do QGIS. Durante a execução, o script processou as instruções e carregou o layout **MODELO_PADRAO_PAISAGEM** previamente configurado no Gerenciador de Layouts do QGIS. Em seguida, o Mapa Principal foi configurado com base no grupo e na camada de referência especificados, ajustando automaticamente a extensão para que o conteúdo se encaixasse perfeitamente dentro da moldura do mapa, sem distorções. O Mapa de Localização também foi configurado com base no grupo auxiliar e na camada de referência definida, garantindo a visualização adequada da área de localização.

Ao final da execução, o layout foi gerado com sucesso e apareceu no Gerenciador de Layouts do QGIS com o nome **Mapa_Gerado**. No layout, o Mapa Principal exibiu os dados do grupo **MAPA PRINCIPAL**, ajustados corretamente à moldura, enquanto o Mapa de Localização apresentou as informações do grupo **MAPA DE LOCALIZAÇÃO**, destacando a área de interesse especificada pela camada de referência. O mapa gerado estava pronto para

ajustes finais, exportação e utilização, atendendo aos padrões esperados de qualidade e eficiência para o projeto. (Figura 23). (Figura 24).

Figura 23 - Demonstração da Confirmação do Funcionamento do Código.



```

1 # Python Console
2 # Use if/else to access QGIS API interface or type '?' to
3 # show info
4 >>> exec(Fetch('C:/Users/NEUCOM-1/AppData/Local/Temp/tmp
5 # Layout 'Mapa_Gerado_novo' foi configurado e está disp
6
>>>

```

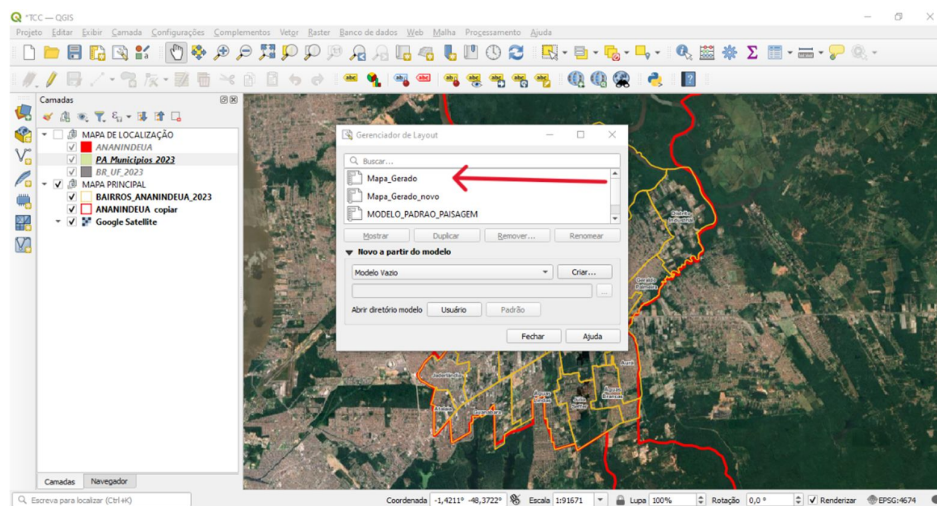
```

127
128
129
130
131 # Localizar a camada de referência para o Mapa de Localização
132 camada_localizacao = projeto.mapLayersByName(camada_referencia_localizacao)
133 if not camada_localizacao:
134     raise Exception(f"A camada de referência '{camada_referencia_localizacao}' não foi enco
135     extent_localizacao = camada_localizacao[0].extent()
136     mapa2.setExtent(extent_localizacao)
137
138 # Informar que o layout foi configurado com sucesso
139 print(f"O layout '{layout_nome}' foi configurado e está disponível no Gerenciador de Layout
140
141 except Exception as e:
142     print(f"Erro ao gerar o mapa: {e}")
143
144
145 # Configurações do usuário
146 layout_modelo_escolhido = "MODELO_PADRAO_PAISAGEM" # Alternativas: MODELO_PADRAO_RETRATO
147 grupo_mapa_principal = "MAPA_PRINCIPAL" # Nome do grupo principal
148 grupo_mapa_localizacao = "MAPA DE LOCALIZAÇÃO" # Nome do grupo de localização
149 camada_referencia_principal = "ANANINDEUA_copiar" # Nome da camada de referência para o Mapa Prin
150 camada_referencia_localizacao = "PA_Municipios_2023" # Nome da camada de referência para o Mapa de
151 nome_layout_novo = "Mapa_Gerado"
152
153 # Executar a função
154 gerar_mapa_com_modelo(
155     nome_layout_novo,
156     grupo_mapa_principal,
157     grupo_mapa_localizacao,
158     camada_referencia_principal,
159     camada_referencia_localizacao,
160     layout_modelo_escolhido

```

Fonte: Desenvolvido pelo Autor.

Figura 24 - Confirmação do Layout Criado no Gerenciador.

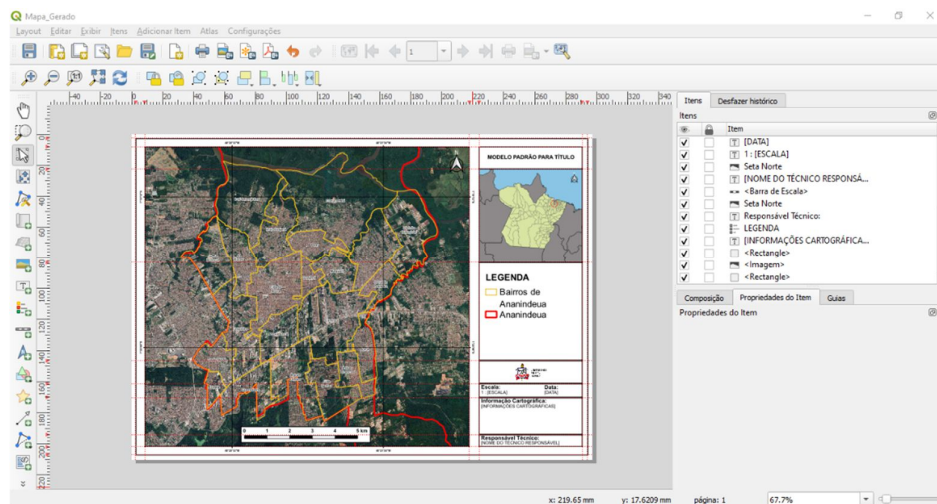


Fonte: Desenvolvido pelo Autor.

Com base no teste realizado, foi confirmado que o código funciona corretamente e cumpre seu objetivo de automatizar a criação de mapas padronizados no QGIS. Ele demonstrou ser uma ferramenta prática e eficiente, capaz de simplificar o trabalho com

layouts complexos, especialmente em projetos acadêmicos como TCCs ou em contextos profissionais que exigem padronização e produtividade. (Figura 25).

Figura 25 - Resultado final e Criação do Layout Adaptado para as Camadas Escolhidas.



Fonte: Desenvolvido pelo Autor.

5.4 Automatização Da Análise Espacial No Qgis: Desenvolvimento De Um Código Para Avaliação Temporal De Infraestruturas Urbanas E Rurais No Sea Geo

O **SEA GEO** desempenha um papel fundamental na análise espacial e temporal de infraestruturas essenciais em áreas urbanas e rurais. A automatização desse processo por meio do **QGIS** permite a coleta, organização e interpretação de dados de maneira mais eficiente e precisa. Isso possibilita um monitoramento detalhado de postos de saúde, lazer, educação, pontos religiosos, residenciais e comerciais ao longo do tempo, fornecendo informações valiosas para gestores públicos, pesquisadores e planejadores urbanos. Dessa forma, o sistema contribui para um planejamento mais estratégico e embasado, garantindo uma melhor qualidade de vida para a população.

A importância desse estudo para a pesquisa acadêmica e aplicada é significativa, pois permite avaliar tendências de crescimento, identificar áreas de vulnerabilidade e sugerir soluções baseadas em dados concretos. A análise temporal realizada pelo **SEA GEO** possibilita uma visão clara das transformações em determinada região ao longo dos anos, evidenciando padrões de desenvolvimento ou degradação. Essa abordagem fortalece a tomada

de decisão, permitindo que políticas públicas e investimentos sejam direcionados de maneira mais eficiente e assertiva.

Além disso, a integração do **SEA GEO** com um código automatizado no **QGIS** representa um avanço tecnológico para a análise geoespacial. O uso de ferramentas de geoprocessamento reduz o tempo necessário para a análise dos dados e minimiza erros humanos, tornando os resultados mais confiáveis. Isso é especialmente relevante para estudos que envolvem grandes volumes de informações espaciais, garantindo que os dados sejam interpretados com maior precisão e consistência ao longo do tempo.

Por fim, essa pesquisa contribui significativamente para o aprimoramento dos métodos de análise espacial e a modernização das ferramentas de planejamento territorial. A automatização da avaliação de infraestruturas urbanas e rurais no **SEA GEO** não só melhora a eficiência dos estudos acadêmicos e institucionais, mas também proporciona um impacto direto na formulação de políticas públicas mais eficazes. Dessa maneira, a pesquisa se torna uma referência para futuras iniciativas na área de geoprocessamento e planejamento territorial, promovendo um desenvolvimento mais sustentável e equilibrado das cidades e do meio rural.

5.4.1 Apresentação Dos Tópicos Avaliados No Sea Geo

O **SEA GEO** realiza a análise espacial e temporal de diferentes infraestruturas essenciais para a qualidade de vida da população, tanto em áreas urbanas quanto rurais. Esses pontos são avaliados com base em critérios específicos, permitindo um monitoramento contínuo das condições e da evolução dos serviços e estruturas de cada região. A partir dessa análise, é possível compreender o impacto dessas infraestruturas no desenvolvimento social e econômico, além de subsidiar políticas públicas e investimentos estratégicos.

Os tópicos analisados incluem **postos de saúde, pontos educacionais, postos de lazer, pontos religiosos, pontos residenciais e pontos comerciais**. Cada um desses elementos desempenha um papel fundamental na organização das cidades e comunidades rurais, influenciando diretamente o bem-estar e a qualidade de vida dos moradores. A disponibilidade e a qualidade desses serviços afetam diversos aspectos do cotidiano, como o acesso à saúde e à educação, o desenvolvimento econômico, a oferta de moradia e a promoção do convívio social.

A análise temporal desses pontos permite identificar padrões de crescimento ou degradação ao longo dos anos. Regiões onde há melhorias em infraestrutura, como a

construção de novos postos de saúde ou escolas, tendem a apresentar avanços nos indicadores sociais e econômicos. Por outro lado, áreas onde esses serviços estão em declínio podem enfrentar desafios como aumento da desigualdade, dificuldades de acesso a serviços básicos e queda na qualidade de vida.

Dessa forma, o estudo desses tópicos no **SEA GEO** oferece uma visão detalhada sobre o desenvolvimento das regiões analisadas. Ao automatizar esse processo por meio do **QGIS**, é possível tornar a coleta e interpretação de dados mais eficiente e precisa, auxiliando tanto na pesquisa acadêmica quanto na formulação de políticas públicas. A seguir, serão abordados detalhadamente cada um desses pontos e sua importância para o planejamento territorial.

5.4.2 Critérios Para Atribuição De Notas No Sea Geo

O **SEA GEO** utiliza um sistema de pontuação de 0 a 5 para avaliar a qualidade e disponibilidade dos diferentes tipos de infraestrutura em uma determinada área. Esse critério permite mensurar o nível de desenvolvimento e as condições dos serviços analisados, fornecendo um diagnóstico detalhado das melhorias ou retrocessos observados ao longo do tempo. A nota é atribuída com base em aspectos como acessibilidade, estado de conservação, quantidade de equipamentos disponíveis e capacidade de atendimento da demanda local.

- **Nota 0-1 (Crítico):** Representa uma situação muito precária ou inexistente. Por exemplo, um posto de saúde que recebe essa pontuação pode estar fechado, sem atendimento adequado ou com infraestrutura em estado crítico. Da mesma forma, um ponto educacional com essa nota pode indicar a falta de escolas na região ou unidades sem condições mínimas de funcionamento.
- **Nota 2-3 (Intermediário):** Indica a presença do serviço ou infraestrutura, mas com limitações que comprometem seu funcionamento ideal. Pode significar que um posto de saúde possui poucos profissionais e equipamentos, que uma escola carece de estrutura adequada para todos os alunos ou que um comércio não atende suficientemente à demanda local. Essa pontuação sugere que a infraestrutura existe, mas precisa de melhorias para atender plenamente à população.
- **Nota 4-5 (Adequado a Excelente):** Indica boa ou excelente qualidade da infraestrutura ou serviço. Locais com essa pontuação possuem acesso adequado, estrutura bem conservada e capacidade suficiente para atender a demanda da população. Um ponto de lazer com nota 5, por exemplo, pode ser um parque bem

cuidado, com segurança, iluminação e manutenção frequente, enquanto um ponto educacional bem avaliado indica escolas bem equipadas e com bom desempenho educacional.

A aplicação dessas notas ao longo de dois anos permite que pesquisadores e gestores identifiquem tendências de melhoria, estagnação ou degradação nos diferentes serviços avaliados. Assim, a análise do SEA GEO se torna uma ferramenta essencial para monitorar o desenvolvimento urbano e rural, subsidiando a tomada de decisões para investimentos e políticas públicas mais eficientes.

5.4.3 Estrutura Do Código Em Python

O código é estruturado de forma modular, o que significa que cada parte tem uma função específica para garantir que o processo de criação do mapa ocorra corretamente no QGIS. Abaixo, explicamos cada trecho do código, detalhando seu funcionamento e objetivo. (Figura 26).

Figura 26 - Bloco das Bibliotecas para o Funcionamento do Código do SEA GEO.

```

1 = from qgis.core import (
2     QgsProject,
3     QgsPrintLayout,
4     QgsLayoutItemMap,
5     QgsCoordinateReferenceSystem,
6     QgsReadWriteContext,
7     QgsRectangle,
8 )
9 from PyQt5.QtXml import QDomDocument
10 import os

```

Fonte: Desenvolvido pelo Autor.

Este bloco importa todas as bibliotecas necessárias para o funcionamento do código:

- **QgsProject** – Permite acessar o projeto atual do QGIS.
- **QgsPrintLayout** – Responsável por manipular os layouts do QGIS.
- **QgsLayoutItemMap** – Utilizado para manipular elementos de mapas dentro do layout.
- **QgsCoordinateReferenceSystem** – Permite trabalhar com sistemas de coordenadas.
- **QgsReadWriteContext** – Usado para carregar layouts salvos como arquivos XML.

- **QgsRectangle** – Representa a extensão geográfica de um mapa.
- **QDomDocument** – Lida com a manipulação de arquivos XML, que são usados para os layouts do QGIS.
- **os** – Biblioteca para manipulação de arquivos e caminhos no sistema operacional. (Figura 27).

Figura 27 - Função para Ajustar a Extensão do Mapa no SEA GEO.

```

13 = def .ajustar_extensao_para_moldura (extent, .mapa):
14 =     """
15     Ajusta a extensão para caber dentro da moldura do mapa, preservando as proporções.
16
17     Args:
18     | extent (QgsRectangle): Extensão original.
19     | mapa (QgsLayoutItemMap): Objeto do mapa no layout.
20
21     Returns:
22     | QgsRectangle: Extensão ajustada.
23     """

```

Fonte: Desenvolvido pelo Autor.

Esta função ajusta a extensão do Mapa Principal para que ele se encaixe corretamente na moldura do layout, mantendo as proporções originais do mapa. (Figura 28).

Figura 28 - Função para Dimensão do Mapa no SEA GEO.

```

24     largura_moldura = .mapa.rect().width()
25     altura_moldura = .mapa.rect().height()
26
27     largura_extent = .extent.width()
28     altura_extent = .extent.height()
29     razao_aspecto_extent = .largura_extent / .altura_extent
30
31     razao_aspecto_moldura = .largura_moldura / .altura_moldura

```

Fonte: Desenvolvido pelo Autor.

Neste trecho do código, são obtidas as dimensões da moldura que contém o mapa dentro do layout, um procedimento essencial para assegurar que a imagem não sofra cortes ou distorções indesejadas. Além disso, é realizada a determinação da razão entre a largura e a altura do mapa, um parâmetro fundamental para preservar a integridade da representação cartográfica ao ajustar a extensão da visualização dentro dos limites da moldura. Por fim, é calculada a proporção da moldura do layout, permitindo a comparação com a proporção

original do mapa, a fim de garantir a fidelidade da exibição e evitar qualquer tipo de deformação na imagem final. (Figura 29).

Figura 29 - Função de Altura e Comprimento do Mapa SEA GEO.

```
33 =     if .razao_aspecto_extent .> .razao_aspecto_moldura:
34         nova_altura = .largura_extent / .razao_aspecto_moldura
35         centro_y = .extent.center().y()
36         extent.setYMinimum(centro_y - .nova_altura / .2)
37         extent.setYMaximum(centro_y + .nova_altura / .2)
38 =     else:
39         nova_largura = .altura_extent * .razao_aspecto_moldura
40         centro_x = .extent.center().x()
41         extent.setXMinimum(centro_x - .nova_largura / .2)
42         extent.setXMaximum(centro_x + .nova_largura / .2)
43
44     return .extent
45
```

Fonte: Desenvolvido pelo Autor.

Neste segmento do código, realiza-se a verificação da relação entre a extensão do mapa e as dimensões da moldura do layout, a fim de determinar se a largura do mapa excede a da moldura. Caso essa condição seja satisfeita, o código procede ao ajuste da altura do mapa, garantindo que sua exibição ocorra de maneira proporcional e sem distorções. De modo análogo, caso a altura do mapa seja superior à da moldura, o algoritmo ajusta a largura, assegurando que o conteúdo cartográfico seja inteiramente visível dentro dos limites estabelecidos, sem cortes ou perda de informações relevantes. Após a aplicação desses ajustes, a função retorna a nova extensão devidamente recalibrada, permitindo sua correta utilização no layout. (Figura 30). (Figura 31).

Figura 30 - Funções para Gerar o Mapa SEA GEO – Parte 1.

```

47 def gerar_mapa_sea_geo(layout_nome, layout_modelo):
48     """
49     Gera um mapa para o SEA GEO usando o layout modelo especificado.
50
51     Args:
52         layout_nome (str): Nome do layout a ser criado.
53         layout_modelo (str): Nome do layout modelo para usar (PAISAGEM ou RETRATO).
54     """
55     try:
56         projeto = QgsProject.instance()
57         layout_manager = projeto.layoutManager()
58
59         layout_existente = layout_manager.layoutByName(layout_modelo)
60         if not layout_existente:
61             raise Exception(f"O layout modelo '{layout_modelo}' não foi encontrado.")
62
63         temp_modelo_path = os.path.join(os.path.expanduser("~"), "temp_layout_modelo.qpt")
64         context = QgsReadWriteContext()
65         if not layout_existente.saveAsTemplate(temp_modelo_path, context):
66             raise Exception("Erro ao exportar o modelo do layout existente para arquivo.")
67
68         dom_document = QDomDocument()
69         with open(temp_modelo_path, "r", encoding="utf-8") as file:
70             dom_document.setContent(file.read())
71
72         layout = QgsPrintLayout(projeto)
73         layout.initializeDefaults()
74         if not layout.loadFromTemplate(dom_document, context):
75             raise Exception("Erro ao carregar o modelo no novo layout.")
76
77         existing_layouts = [l.name() for l in layout_manager.layouts()]
78         if layout_nome in existing_layouts:
79             layout_nome += "_novo"

```

Fonte: Desenvolvido pelo Autor.

Figura 31 - Funções para Gerar o Mapa SEA GEO – Parte 2.

```

81         layout.setName(layout_nome)
82         layout_manager.addLayout(layout)
83
84         # Configurar o Mapa Principal (SEA)
85         mapa1 = layout.itemById("Mapa 1")
86         if not mapa1 or not isinstance(mapa1, QgsLayoutItemMap):
87             raise Exception("O layout modelo não contém um item 'Mapa 1' válido.")
88
89         grupo_mapa_principal = "SEA"
90         camada_referencia_principal = projeto.mapLayersByName(grupo_mapa_principal)
91         if not camada_referencia_principal:
92             raise Exception(f"O grupo '{grupo_mapa_principal}' não foi encontrado.")
93
94         extent_principal = camada_referencia_principal[0].extent()
95         extent_ajustado = ajustar_extensao_para_moldura(extent_principal, mapa1)
96         mapa1.setExtent(extent_ajustado)
97
98         # Configurar o Mapa de Localização (MAPA DE LOCALIZAÇÃO - SEA)
99         mapa2 = layout.itemById("Mapa 2")
100        if not mapa2 or not isinstance(mapa2, QgsLayoutItemMap):
101            raise Exception("O layout modelo não contém um item 'Mapa 2' válido.")
102
103        grupo_mapa_localizacao = "MAPA DE LOCALIZAÇÃO - SEA"
104        camada_referencia_localizacao = projeto.mapLayersByName(grupo_mapa_localizacao)
105        if not camada_referencia_localizacao:
106            raise Exception(f"O grupo '{grupo_mapa_localizacao}' não foi encontrado.")
107
108        extent_localizacao = camada_referencia_localizacao[0].extent()
109        mapa2.setExtent(extent_localizacao)
110
111        print(f"O layout '{layout_nome}' foi configurado e está disponível no Gerenciador de Layouts.")
112
113    except Exception as e:
114        print(f"Erro ao gerar o mapa: {e}")

```

Fonte: Desenvolvido pelo Autor.

A presente função tem por objetivo a criação e configuração do **Mapa Principal** e do **Mapa de Localização** dentro do ambiente de layout do software **QGIS**. Sua execução assegura o correto posicionamento das camadas nos respectivos mapas, ajustando suas extensões e escalas de maneira automatizada. Dessa forma, possibilita-se a geração de mapas padronizados sem a necessidade de intervenção manual para a configuração individual de cada elemento compositivo do layout.

Inicialmente, o código obtém a instância ativa do projeto no **QGIS** e acessa o **Gerenciador de Layouts**, o qual centraliza e organiza todos os layouts disponíveis dentro do projeto. Este gerenciador desempenha um papel fundamental na criação, modificação e estruturação dos layouts, garantindo que um novo layout possa ser corretamente gerado e disponibilizado para o usuário sem comprometimentos na organização do projeto.

Na sequência, o código verifica a existência prévia do layout modelo especificado, podendo este estar configurado no formato **PAISAGEM** ou **RETRATO**. Caso o modelo desejado não esteja presente no projeto, uma mensagem de erro é exibida para alertar o usuário sobre a ausência do referido layout. Tal verificação previne que o código tente acessar um modelo inexistente, evitando falhas no processamento e garantindo a estabilidade da execução do script.

Para preservar a integridade do layout original, o código salva temporariamente o modelo de layout como um arquivo no formato **.qpt (QGIS Print Template)**. Esse arquivo atua como uma cópia temporária do layout original, permitindo que o código manipule e realize alterações no modelo sem comprometer sua estrutura primária. Dessa forma, evita-se modificações diretas no modelo de referência, assegurando sua preservação para usos futuros.

Uma vez salvo, o arquivo **.qpt** é carregado e interpretado como um documento no formato **XML**, possibilitando a edição dinâmica do layout antes da criação do novo mapa. No **QGIS**, os layouts são armazenados em uma estrutura baseada em **XML**, o que permite a leitura programática e a modificação de seus atributos. Essa abordagem viabiliza a incorporação de novas configurações ao modelo antes da geração do novo layout, tornando o processo altamente flexível e adaptável às necessidades específicas do usuário.

A etapa subsequente envolve a criação do novo layout dentro do **QGIS**, utilizando como base o modelo salvo no arquivo **.qpt**. Esse procedimento assegura que todas as configurações do layout original sejam preservadas, incluindo o posicionamento dos elementos gráficos, as escalas e demais propriedades visuais. Dessa forma, o usuário pode gerar um novo layout automaticamente, sem necessidade de realizar ajustes manuais extensivos.

Antes de integrar o novo layout ao projeto, o código executa uma verificação para identificar a existência de layouts com o mesmo nome dentro do **Gerenciador de Layouts**. Caso seja detectado um layout previamente nomeado de maneira idêntica, o código aplica automaticamente o sufixo "**_novo**" ao nome do novo layout, evitando conflitos e garantindo que cada instância gerada possua uma denominação única.

Por fim, o novo layout é devidamente nomeado e incorporado ao **Gerenciador de Layouts** do **QGIS**, tornando-se acessível para visualização e edição. O usuário pode consultá-lo diretamente no gerenciador, revisar as configurações aplicadas, realizar ajustes finais e proceder com a exportação do mapa nos formatos **PDF** ou **imagem**, conforme a necessidade específica do projeto. (Figura 32).

Figura 32 - Configuração do Usuário no Mapa SEA GEO.

```

117 #·Configuração·do·usuário
118 layout_modelo_escolhido = "MODELO_PADRAO_SEA_GEO" ·#·Alternativas:·MODELO_PADRAO_RETRATO
119 nome_layout_novo = "Mapa_SEA_GEO"
120
121 #·Executar·a·função
122 gerar_mapa_sea_geo(nome_layout_novo, layout_modelo_escolhido)

```

Fonte: Desenvolvido pelo Autor.

Neste trecho do código, realiza-se a busca e identificação do **Mapa Principal** dentro do layout do **QGIS**. O algoritmo localiza o item correspondente no layout previamente carregado, assegurando que todas as configurações necessárias possam ser aplicadas de forma adequada. O **Mapa Principal** constitui o elemento central do layout cartográfico, sendo a área destinada à exibição das informações mais relevantes do projeto.

Na sequência, o código procede à identificação das camadas contidas no grupo denominado "SEA", cuja função é definir a extensão espacial do **Mapa Principal**. Esse grupo engloba todas as camadas que devem ser representadas na composição cartográfica, sendo que uma camada específica dentro desse conjunto é utilizada como referência para estabelecer os limites espaciais do mapa dentro da moldura do layout.

Após a determinação da camada de referência, o código executa o ajuste da escala do **Mapa Principal**, garantindo que sua exibição se adeque corretamente às dimensões da moldura do layout. Esse ajuste é realizado por meio da análise da extensão espacial da camada, de forma a redimensioná-la proporcionalmente. Tal procedimento assegura a visibilidade integral dos dados, prevenindo cortes ou distorções que possam comprometer a interpretação das informações cartográficas.

Posteriormente, o código direciona-se à busca do **Mapa de Localização** dentro do layout e identifica as camadas pertencentes ao grupo "**MAPA DE LOCALIZAÇÃO - SEA**". Esse grupo contém camadas auxiliares cuja finalidade é proporcionar um contexto

espacial ao **Mapa de Localização**, permitindo ao usuário visualizar a posição relativa do **Mapa Principal** em relação a uma área geográfica mais abrangente.

Uma vez localizadas as camadas, o código define automaticamente a extensão espacial do **Mapa de Localização** com base no conteúdo desse grupo. O ajuste da área exibida dentro da moldura é realizado de maneira automatizada, garantindo que a visão geral da área de estudo seja corretamente representada dentro do layout do **QGIS**.

Por fim, o código exibe uma mensagem de confirmação indicando que o layout foi gerado com êxito. Esse aviso sinaliza que todas as etapas do processo foram concluídas satisfatoriamente e que o novo layout se encontra disponível no **Gerenciador de Layouts** do **QGIS**, possibilitando sua visualização, edição e eventual exportação nos formatos desejados pelo usuário.

5.4.4 Como Utilizar O Código No Qgis

O código foi desenvolvido para ser simples e intuitivo de utilizar dentro do **QGIS**, permitindo que qualquer usuário, mesmo sem experiência em programação, possa gerar mapas de forma automática e padronizada. Para garantir que o código funcione corretamente, é essencial seguir alguns passos para preparar o ambiente dentro do **QGIS**.

- **Organizar as Camadas no QGIS**

Antes de rodar o código, é fundamental que as camadas estejam organizadas corretamente dentro do Painel de Camadas do **QGIS**. O código foi projetado para buscar automaticamente as camadas dentro de dois grupos específicos, por isso, é necessário que esses grupos estejam criados e contenham as informações corretas.

- **Estrutura necessária no QGIS**

- SEA → Este grupo deve conter todas as camadas que farão parte do Mapa Principal. Ele pode incluir arquivos shapefile (.shp), camadas raster (.tif), camadas vetoriais ou qualquer outro dado geoespacial relevante para o mapa principal.
- MAPA DE LOCALIZAÇÃO - SEA → Esse grupo deve conter as camadas auxiliares para o Mapa de Localização, que servem para fornecer um contexto espacial ao mapa principal. Ele pode conter uma camada base com os limites administrativos da região, estradas, rios ou outras informações geográficas importantes. (Figura 33).

Figura 33 - Formação dos Grupos para o Funcionamento do Mapa SEA GEO.



Fonte: Desenvolvido pelo Autor.

Caso esses grupos não existam ou estejam vazios, o código não conseguirá identificar as camadas e não será possível gerar o mapa corretamente.

- **Passo a passo para organizar as camadas**
 - No Painel de Camadas do QGIS, clique com o botão direito e selecione Adicionar Grupo.
 - Nomeie o primeiro grupo como SEA e insira dentro dele as camadas que deseja visualizar no Mapa Principal.
 - Crie um segundo grupo chamado MAPA DE LOCALIZAÇÃO - SEA e adicione dentro dele as camadas que devem ser exibidas no Mapa de Localização.
 - Verifique se todas as camadas estão visíveis e carregadas corretamente no QGIS antes de executar o código.

5.4.5 Código Para Automação Do Mapa Do Sea Geo Completo (Anexo 4) (Figura 34). (Figura 35). (Figura 36). (Figura 37).

Figura 34 - Código Completo do Mapa SEA GEO – Parte 1.

```

1 from qgis.core import (
2     QgsProject,
3     QgsPrintLayout,
4     QgsLayoutItemMap,
5     QgsCoordinateReferenceSystem,
6     QgsReadWriteContext,
7     QgsRectangle,
8 )
9 from PyQt5.QtXml import QDomDocument
10 import os
11
12
13 def ajustar_extensao_para_moldura(extent, mapa):
14     """
15     Ajusta a extensão para caber dentro da moldura do mapa, preservando as proporções.
16
17     Args:
18         extent (QgsRectangle): Extensão original.
19         mapa (QgsLayoutItemMap): Objeto do mapa no layout.
20
21     Returns:
22         QgsRectangle: Extensão ajustada.
23     """
24     largura_moldura = mapa.rect().width()
25     altura_moldura = mapa.rect().height()
26
27     largura_extent = extent.width()
28     altura_extent = extent.height()
29     razao_aspecto_extent = largura_extent / altura_extent
30
31     razao_aspecto_moldura = largura_moldura / altura_moldura

```

Fonte: Desenvolvido pelo Autor.

Figura 35 - Código Completo do Mapa SEA GEO – Parte 2.

```

33 if razao_aspecto_extent > razao_aspecto_moldura:
34     nova_altura = largura_extent / razao_aspecto_moldura
35     centro_y = extent.center().y()
36     extent.setMinimum(centro_y - nova_altura / 2)
37     extent.setMaximum(centro_y + nova_altura / 2)
38 else:
39     nova_largura = altura_extent * razao_aspecto_moldura
40     centro_x = extent.center().x()
41     extent.setMinimum(centro_x - nova_largura / 2)
42     extent.setMaximum(centro_x + nova_largura / 2)
43
44     return extent
45
46
47 def gerar_mapa_sea_geo(layout_nome, layout_modelo):
48     """
49     Gera um mapa para o SEA GEO usando o layout modelo especificado.
50
51     Args:
52         layout_nome (str): Nome do layout a ser criado.
53         layout_modelo (str): Nome do layout modelo para usar (PAISAGEM ou RETRATO).
54     """
55     try:
56         projeto = QgsProject.instance()
57         layout_manager = projeto.layoutManager()
58
59         layout_existente = layout_manager.layoutByName(layout_modelo)
60         if not layout_existente:
61             raise Exception(f"O layout modelo '{layout_modelo}' não foi encontrado.")
62
63         temp_modelo_path = os.path.join(os.path.expanduser("~"), "temp_layout_modelo.qpt")
64         context = QgsReadWriteContext()
65         if not layout_existente.saveAsTemplate(temp_modelo_path, context):
66             raise Exception("Erro ao exportar o modelo do layout existente para arquivo.")

```

Fonte: Desenvolvido pelo Autor.

Figura 36 - Código Completo do Mapa SEA GEO – Parte 3.

```

68 dom_document = QDomDocument()
69 with open(temp_modelo_path, "r", encoding="utf-8") as file:
70     dom_document.setContent(file.read())
71
72 layout = QgsPrintLayout(projeto)
73 layout.initializeDefaults()
74 if not layout.loadFromTemplate(dom_document, context):
75     raise Exception("Erro ao carregar o modelo no novo layout.")
76
77 existing_layouts = [l.name() for l in layout_manager.layouts()]
78 if layout_nome in existing_layouts:
79     layout_nome += "_novo"
80
81 layout.setName(layout_nome)
82 layout_manager.addLayout(layout)
83
84 # Configurar o Mapa Principal (SEA)
85 mapal = layout.itemById("Mapa 1")
86 if not mapal or not isinstance(mapal, QgsLayoutItemMap):
87     raise Exception("O layout modelo não contém um item 'Mapa 1' válido.")
88
89 grupo_mapa_principal = "SEA"
90 camada_referencia_principal = projeto.mapLayersByName(grupo_mapa_principal)
91 if not camada_referencia_principal:
92     raise Exception(f"O grupo '{grupo_mapa_principal}' não foi encontrado.")
93
94 extent_principal = camada_referencia_principal[0].extent()
95 extent_ajustado = ajustar_extensao_para_moldura(extent_principal, mapal)
96 mapal.setExtent(extent_ajustado)
97
98 # Configurar o Mapa de Localização (MAPA DE LOCALIZAÇÃO - SEA)
99 mapa2 = layout.itemById("Mapa 2")
100 if not mapa2 or not isinstance(mapa2, QgsLayoutItemMap):
101     raise Exception("O layout modelo não contém um item 'Mapa 2' válido.")

```

Fonte: Desenvolvido pelo Autor.

Figura 37 - Código Completo do Mapa SEA GEO – Parte 4.

```

89 grupo_mapa_principal = "SEA"
90 camada_referencia_principal = projeto.mapLayersByName(grupo_mapa_principal)
91 if not camada_referencia_principal:
92     raise Exception(f"O grupo '{grupo_mapa_principal}' não foi encontrado.")
93
94 extent_principal = camada_referencia_principal[0].extent()
95 extent_ajustado = ajustar_extensao_para_moldura(extent_principal, mapal)
96 mapal.setExtent(extent_ajustado)
97
98 # Configurar o Mapa de Localização (MAPA DE LOCALIZAÇÃO - SEA)
99 mapa2 = layout.itemById("Mapa 2")
100 if not mapa2 or not isinstance(mapa2, QgsLayoutItemMap):
101     raise Exception("O layout modelo não contém um item 'Mapa 2' válido.")
102
103 grupo_mapa_localizacao = "MAPA DE LOCALIZAÇÃO - SEA"
104 camada_referencia_localizacao = projeto.mapLayersByName(grupo_mapa_localizacao)
105 if not camada_referencia_localizacao:
106     raise Exception(f"O grupo '{grupo_mapa_localizacao}' não foi encontrado.")
107
108 extent_localizacao = camada_referencia_localizacao[0].extent()
109 mapa2.setExtent(extent_localizacao)
110
111 print(f"O layout '{layout_nome}' foi configurado e está disponível no Gerenciador de Layouts.")
112
113 except Exception as e:
114     print(f"Erro ao gerar o mapa: {e}")
115
116
117 # Configuração do usuário
118 layout_modelo_escolhido = "MODELO_PADRAO_SEA_GEO" # Alternativas: MODELO_PADRAO_RETRATO
119 nome_layout_novo = "Mapa_SEA_GEO"
120
121 # Executar a função
122 gerar_mapa_sea_geo(nome_layout_novo, layout_modelo_escolhido)

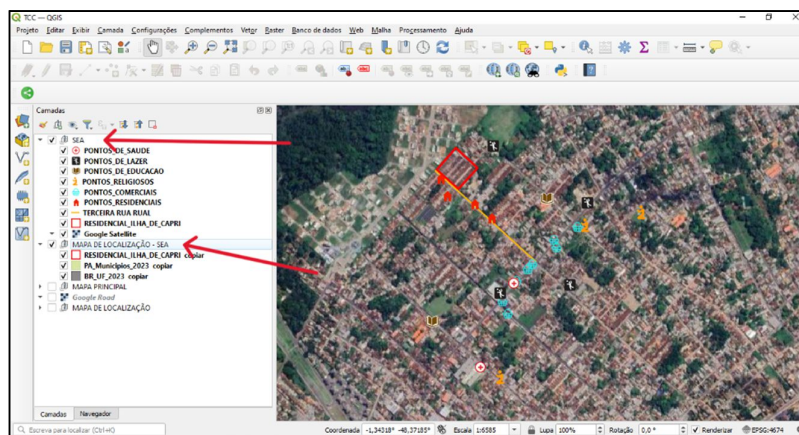
```

Fonte: Desenvolvido pelo Autor.

5.4.6 Teste Do Funcionamento Completo Do Código De Automação Do Sea Geo

No ambiente do **QGIS**, a aba de camadas desempenha um papel fundamental na organização e estruturação dos elementos espaciais dentro do projeto. Durante a execução do código, foi realizada uma verificação criteriosa para garantir a presença das camadas "SEA" e "MAPA DE LOCALIZAÇÃO - SEA", elementos essenciais para a correta configuração do layout cartográfico (Figura 38).

Figura 38 - Verificação dos Grupos "SEA" e "MAPA DE LOCALIZAÇÃO - SEA".



Fonte: Desenvolvido pelo Autor.

Essa etapa é crucial para assegurar que os dados geográficos necessários estejam devidamente carregados e acessíveis dentro do Gerenciamento de Camadas, permitindo ao usuário manipular e editar as informações conforme necessário.

Após a validação da presença das camadas, o próximo passo consistiu na seleção de uma delas para iniciar o processo de edição. Durante essa fase, foram criadas três colunas principais na tabela de atributos, cada uma com uma função específica dentro da análise geoespacial:

- **IDENT (Identificação do ponto):** Esta coluna tem a finalidade de atribuir um identificador único a cada ponto geográfico dentro da camada. A identificação padronizada é essencial para garantir a rastreabilidade dos dados e possibilitar análises comparativas entre diferentes períodos ou cenários.
- **2013 (Ano-base para o estudo):** Esta coluna representa um marco temporal escolhido para a análise retrospectiva do fenômeno em estudo. A escolha do ano de 2013 como referência permite comparar dados históricos com projeções futuras, identificando padrões e tendências na distribuição espacial dos elementos analisados.
- **2025 (Ano do estudo atual):** A terceira coluna representa o ano no qual a análise está sendo conduzida, funcionando como ponto de comparação com os dados do ano-base. Essa abordagem permite a realização de estudos evolutivos, verificando mudanças espaciais, impactos ambientais, tendências de uso e ocupação do solo, entre outras aplicações (Figura 39).

Figura 39 - Criação das Colunas na Tabela de Atributo da Camada.

| id | EVENTO | 2013 | 2015 |
|----|--------------------------------------|------|------|
| 1 | Unidade de Saúde - Heliolândia Rural | 2,3 | 3,4 |
| 2 | UPA 24 H - Distrito Industrial | 3,1 | 4,1 |

Fonte: Desenvolvido pelo Autor.

Uma vez concluída a estruturação das camadas e a criação das colunas necessárias, o código foi exportado e executado dentro do terminal **Python do QGIS**. Essa etapa tem como principal objetivo garantir que o código funcione corretamente dentro do ambiente do QGIS, sem erros de sintaxe ou falhas de execução. A exportação do código permite que ele seja testado de maneira independente, possibilitando a identificação e correção de possíveis inconsistências antes de sua aplicação definitiva no projeto (Figura 40).

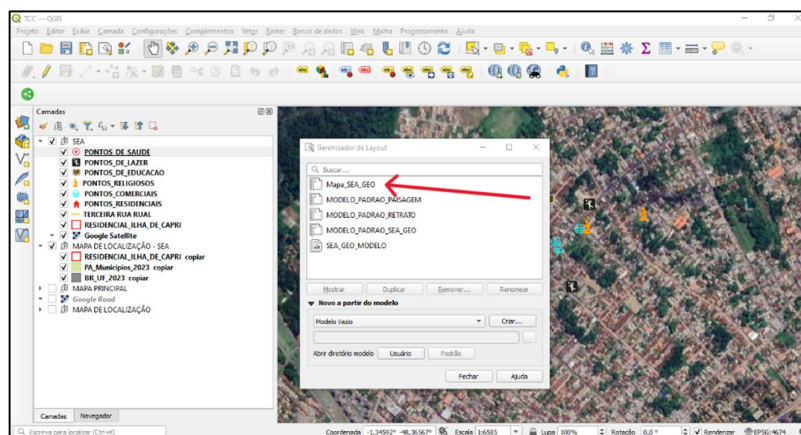
Figura 40 - Verificação da Renomeação Está Correta.

```

1 # Python Console
2 # Use Ctrl+Enter to access QGIS API interface or type '?' for
  more info
3 # Security warning: typing commands from an untrusted s
 ource can harm your computer
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
255
```

estivessem completos e corretamente especificados dentro do projeto. Esse procedimento é essencial para evitar conflitos de nomenclatura e assegurar que os layouts sejam identificáveis e organizados corretamente dentro do **Gerenciamento de Layouts do QGIS**. (Figura 41).

Figura 41 - Verificação se o "Mapa_SEA_GEO" Foi Criada.

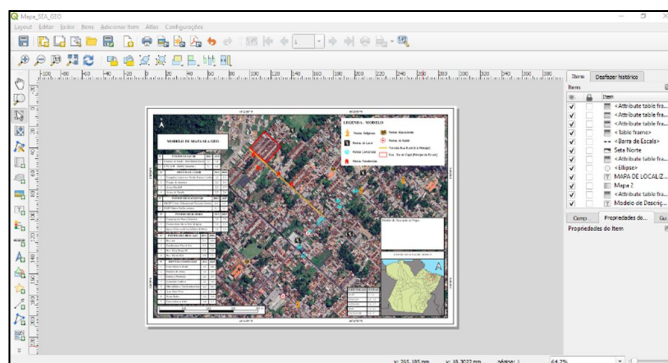


Fonte: Desenvolvido pelo Autor.

Após a conclusão do teste, verificou-se que o layout foi corretamente criado e armazenado no Gerenciamento de Layouts do software. O nome atribuído ao layout foi "**Mapa_SEA_GEO**", confirmando que o modelo foi gerado com sucesso e está disponível para futuras edições e exportações. Esse layout corresponde a um modelo cartográfico, permitindo que novos mapas possam ser gerados automaticamente com base na mesma estrutura, sem necessidade de configurar manualmente cada elemento do layout a cada nova utilização.

A presença do **Mapa Completo** dentro do projeto atesta a eficácia do código na automação do processo de geração de layouts, garantindo que a estrutura cartográfica esteja corretamente configurada e acessível para o usuário. A disponibilidade do layout dentro do **Gerenciador de Layouts** possibilita a visualização direta do mapa, bem como a realização de ajustes finais antes de sua exportação para formatos como **PDF** ou **imagem**, de acordo com as necessidades do projeto. Esse processo otimiza significativamente a produção cartográfica, reduzindo o tempo necessário para a criação de mapas e minimizando possíveis erros manuais durante a configuração do layout (Figura 42).

Figura 42 - Mapa Completo com o Modelo SEA GEO.

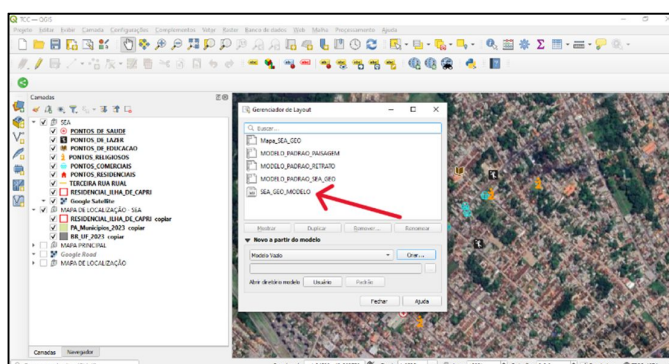


Fonte: Desenvolvido pelo Autor.

5.4.7 Modelo De Relatório Para Complementação Do Mapa Sea Geo

Para a complementação do estudo conduzido em conjunto com o **Mapa SEA GEO**, foi elaborado um relatório anexo, cuja finalidade é expandir e aprofundar a análise dos dados geospaciais representados no mapa. Esse relatório pode ser incorporado diretamente ao projeto como um novo layout, permitindo que seja utilizado de forma integrada dentro do ambiente do QGIS (Figura 43).

Figura 43 - Gerenciamento de Layout Contendo o Relatório.



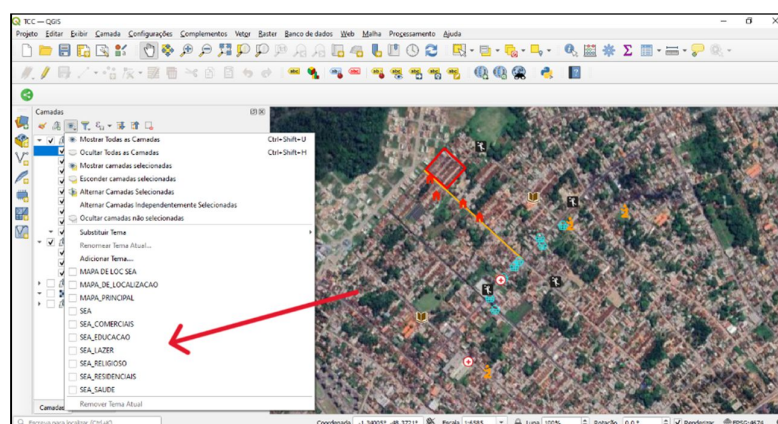
Fonte: Desenvolvido pelo Autor.

A estrutura do relatório deve ser organizada por temas específicos, garantindo a segmentação adequada das informações abordadas na pesquisa. Cada tema será tratado de forma individualizada, permitindo a análise detalhada de diferentes aspectos do espaço

geográfico estudado. Entre os temas que podem ser incluídos no relatório, destacam-se: (Figura 44).

- **Comerciais** – Mapeamento de estabelecimentos comerciais e áreas destinadas a atividades econômicas.
- **Educação** – Distribuição de instituições de ensino, incluindo escolas, universidades e centros de formação profissional.
- **Lazer** – Áreas de recreação e espaços de convivência, como parques, praças e centros culturais.
- **Religioso** – Localização de templos, igrejas, mesquitas e outros espaços destinados a práticas religiosas.
- **Residenciais** – Delimitação de áreas habitacionais, abrangendo diferentes tipos de moradias e suas características.
- **Saúde** – Identificação de hospitais, postos de saúde, clínicas e demais unidades de atendimento médico.
- **SEA** – Categoria que engloba todos os elementos citados anteriormente, permitindo uma visão geral e integrada do espaço analisado.

Figura 44 - Temas Separados Para o Relatório.

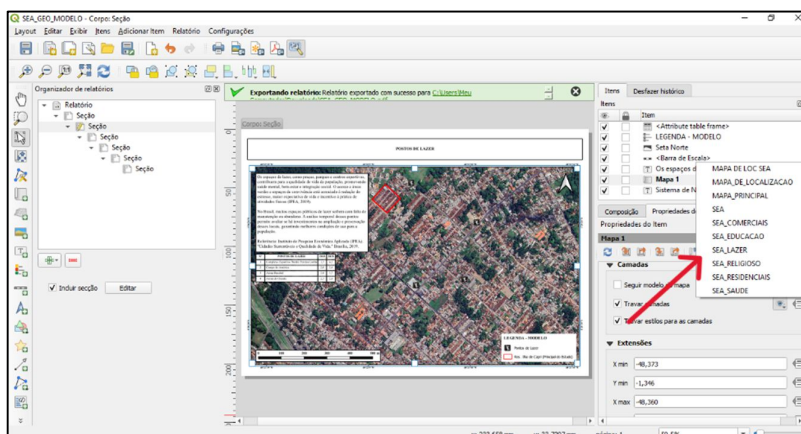


Fonte: Desenvolvido pelo Autor.

Cada um desses temas será associado a um mapa específico, garantindo que as informações geográficas sejam representadas de maneira precisa e contextualizada dentro do estudo. A adaptação do relatório ao projeto cartográfico ocorre por meio da vinculação desses

temas ao processo de geração dos mapas, possibilitando uma análise segmentada e comparativa entre as diferentes categorias representadas. (Figura 45).

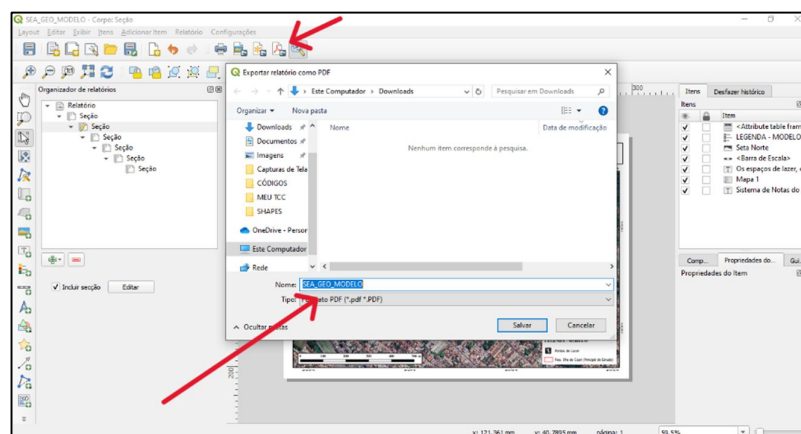
Figura 45 - Atualização do Tema Para Cada Relatório.



Fonte: Desenvolvido pelo Autor.

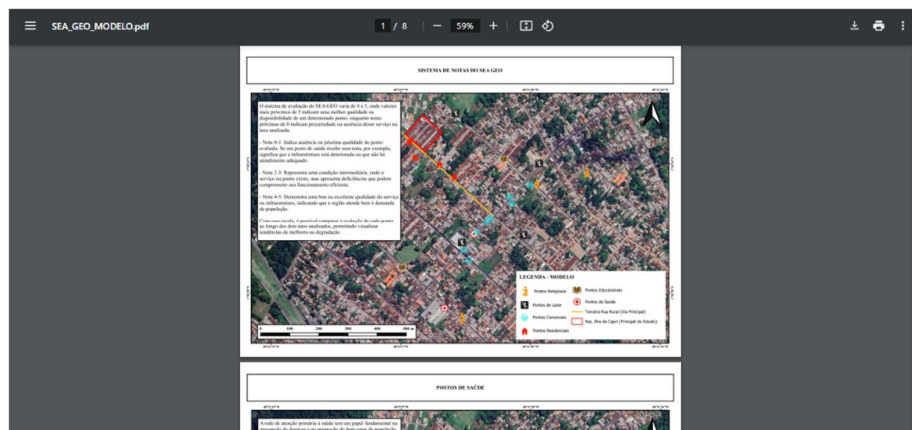
Após a organização e estruturação do relatório dentro do QGIS, o documento será exportado em formato PDF, permitindo sua visualização, compartilhamento e posterior análise. A exportação assegura que todos os elementos do relatório, incluindo textos, gráficos e mapas temáticos, sejam preservados e exibidos de maneira clara e objetiva. (Figura 46).

Figura 46 - Salvamento do Relatório em PDF.



Fonte: Desenvolvido pelo Autor.

Figura 47 - Relatório Finalizado em PDF.

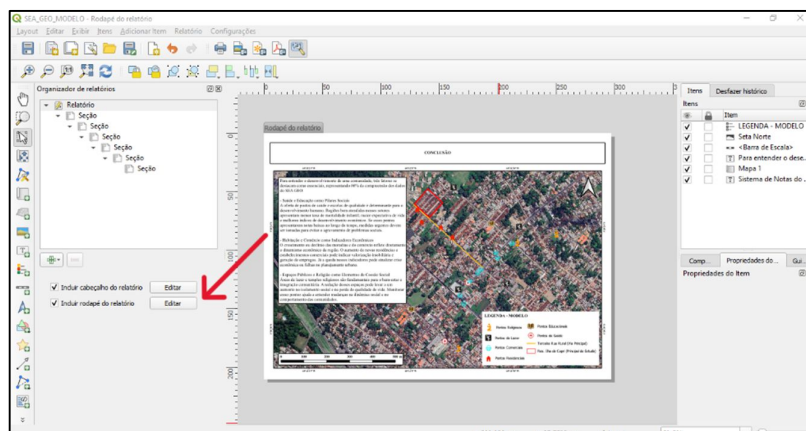


Fonte: Desenvolvido pelo Autor.

Para facilitar a compreensão e a navegação no relatório gerado, a estrutura da aba Relatório no QGIS segue um formato hierárquico, semelhante a uma "escadinha". Nesse modelo, as seções são organizadas da seguinte forma: (Figura 48). (Figura 49).

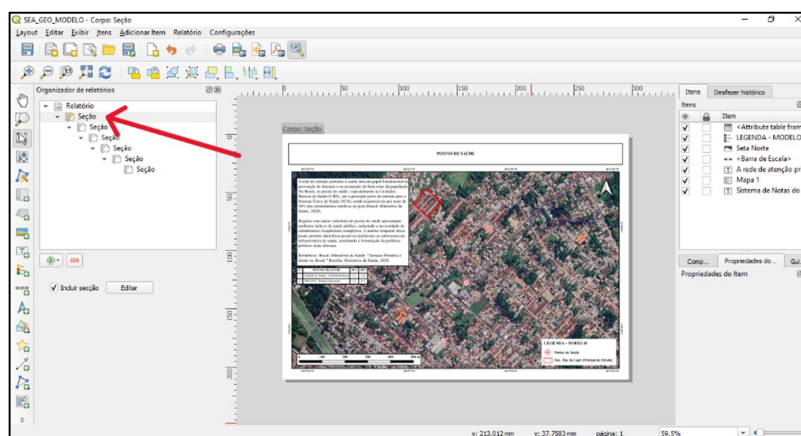
- **Cabeçalho** – Corresponde à primeira página do relatório, onde geralmente são inseridas informações iniciais, como título, autores, data e contexto do estudo.
- **Seções** – Representam as páginas intermediárias do relatório, onde são inseridos os conteúdos analíticos, mapas temáticos e informações segmentadas por tema.
- **Rodapé** – Constitui a última página do relatório, podendo conter considerações finais, referências bibliográficas e anexos adicionais.

Figura 48 - Apresentação do Cabeçalho e Rodapé.



Fonte: Desenvolvido pelo Autor.

Figura 49 - Representação das Seções da Aba Relatório.



Fonte: Desenvolvido pelo Autor.

Essa estruturação hierárquica permite a organização lógica das informações dentro do relatório, facilitando a leitura e a interpretação dos dados apresentados. Dessa forma, o estudo torna-se mais acessível e eficiente, garantindo que as informações espaciais sejam apresentadas de forma clara e objetiva para análise e tomada de decisão.

5.4.8 Modelo Completo Do Relatório (Anexo 5)

Este documento constitui um modelo padrão para a complementação do Mapa SEA GEO e pode ser adaptado conforme a necessidade do estudo. A estrutura apresentada serve como referência, permitindo ajustes nos temas, na organização do conteúdo e na formatação do relatório. Dessa forma, os usuários podem modificar e personalizar o layout de acordo com os objetivos específicos da pesquisa, garantindo que a apresentação dos dados atenda às exigências de cada análise geoespacial.

5.5 Modelos Prontos De Layout Para Todos Os Mapas (Anexo 6):

Os layouts para os códigos serão disponibilizados para uso como modelos de referência, permitindo sua aplicação e adaptação conforme as necessidades específicas de cada projeto. Esses layouts foram estruturados para garantir a organização eficiente dos mapas e relatórios dentro do QGIS, facilitando a padronização da apresentação dos dados. Os

usuários poderão utilizar esses modelos como base, realizando modificações conforme necessário para atender às exigências de suas análises e estudos geoespaciais.

6 A IMPORTÂNCIA DO SEA GEO AMAZÔNICO PARA ANÁLISES TEMPORAIS E SOCIOAMBIENTAIS

O documento **SEA GEO AMAZÔNICO** é uma ferramenta essencial para a análise de dados socioeconômicos e ambientais da região amazônica. Sua estrutura permite a organização de informações relevantes para a compreensão das mudanças que ocorrem ao longo do tempo, auxiliando na formulação de políticas públicas e na gestão dos recursos naturais. A seguir, serão abordados diversos aspectos que destacam a importância desse documento para estudos estratégicos e projeções futuras.

6.1 Monitoramento Das Transformações Socioeconômicas

A análise temporal dos dados apresentados no SEA GEO AMAZÔNICO possibilita um acompanhamento detalhado das mudanças socioeconômicas da região. Isso inclui aspectos como crescimento populacional, desenvolvimento urbano, acesso a serviços essenciais e variações na qualidade de vida das comunidades locais. Essas informações são cruciais para a elaboração de estratégias que visam minimizar desigualdades e promover o desenvolvimento sustentável.

Além disso, o documento auxilia na identificação de padrões de vulnerabilidade social, permitindo que governos e instituições desenvolvam intervenções direcionadas para grupos que enfrentam maiores dificuldades. A observação desses padrões ao longo do tempo possibilita a implementação de medidas mais eficazes, garantindo que os recursos sejam alocados de maneira eficiente.

Outro ponto importante é que a base de dados contida no SEA GEO AMAZÔNICO facilita a realização de estudos comparativos entre diferentes períodos, possibilitando a mensuração do impacto de políticas já implementadas. Isso permite avaliar se determinadas ações surtiram os efeitos desejados ou se ajustes precisam ser feitos para melhorar os resultados.

A capacidade de prever cenários futuros também se fortalece com essa análise temporal. Quando os dados são bem organizados, torna-se viável utilizar modelos estatísticos

para projetar tendências e antecipar possíveis desafios. Essa previsibilidade auxilia na tomada de decisões mais informadas e assertivas.

Além dos aspectos sociais, o documento fornece suporte para pesquisas acadêmicas que buscam compreender a relação entre o crescimento populacional e seus impactos na economia local. Essas análises ajudam a direcionar investimentos para setores-chave, promovendo um crescimento equilibrado e sustentável.

Por fim, o monitoramento contínuo das transformações socioeconômicas possibilita que diferentes atores – incluindo governos, ONGs e empresas privadas – atuem de forma integrada para melhorar as condições de vida da população amazônica. A convergência desses esforços é essencial para garantir um desenvolvimento justo e duradouro.

6.2 Avaliação De Impactos Ambientais E Sustentabilidade

O SEA GEO AMAZÔNICO também desempenha um papel essencial na avaliação dos impactos ambientais causados por atividades humanas na região. O avanço do desmatamento, as mudanças no uso do solo e a degradação de ecossistemas são desafios constantes que precisam ser monitorados e combatidos com base em dados confiáveis.

A presença de informações organizadas e estruturadas permite que cientistas e especialistas avaliem como determinadas atividades econômicas, como mineração e agricultura, afetam o equilíbrio ecológico. Dessa forma, torna-se possível estabelecer políticas que incentivem práticas mais sustentáveis e reduzam a degradação ambiental.

Além disso, a análise temporal dos dados ambientais possibilita uma compreensão mais profunda sobre os efeitos das mudanças climáticas na região amazônica. O aumento da temperatura, a alteração nos padrões de chuvas e a perda de biodiversidade são fenômenos que podem ser monitorados e, conseqüentemente, mitigados por meio de estratégias embasadas em evidências.

Outro benefício da organização dos dados ambientais é a possibilidade de identificar regiões que necessitam de medidas emergenciais de preservação. Com base na análise das informações, pode-se definir áreas prioritárias para conservação e desenvolver programas de reflorestamento e recuperação de ecossistemas degradados.

A importância do SEA GEO AMAZÔNICO também se destaca no contexto das comunidades tradicionais e indígenas que dependem dos recursos naturais para sua subsistência. A proteção desses territórios e a garantia de um equilíbrio ecológico são fundamentais para assegurar a continuidade de suas culturas e modos de vida.

Por fim, a integração das informações ambientais contidas no documento com outros bancos de dados globais permite a construção de estratégias conjuntas de preservação. Organismos internacionais podem utilizar essas informações para planejar ações coordenadas, contribuindo para a proteção da Amazônia em escala mundial.

6.3 Suporte À Formulação De Políticas Públicas

A existência de um banco de dados como o SEA GEO AMAZÔNICO facilita a formulação de políticas públicas mais eficazes e direcionadas às necessidades reais da população. A tomada de decisões embasada em dados concretos é um fator determinante para a eficiência de programas governamentais e ações institucionais.

Os gestores públicos podem utilizar as informações contidas no documento para definir prioridades e otimizar a aplicação de recursos. Com um planejamento adequado, é possível evitar desperdícios e garantir que investimentos sejam direcionados para áreas que realmente precisam de atenção.

Além disso, a análise histórica dos dados permite avaliar os impactos de políticas já implementadas, identificando suas falhas e pontos positivos. Isso possibilita um aprimoramento constante das estratégias governamentais, tornando-as mais eficientes e alinhadas às necessidades da população.

Outro fator relevante é a possibilidade de utilizar as informações para promover ações intersetoriais. A integração entre diferentes setores – como saúde, educação, infraestrutura e meio ambiente – pode ser mais bem coordenada quando há um entendimento claro sobre os desafios e demandas de cada área.

A transparência na gestão pública também é beneficiada pela organização e divulgação desses dados. Quando a população tem acesso às informações coletadas, torna-se mais fácil acompanhar as ações governamentais e cobrar medidas efetivas. Isso fortalece a participação cidadã e a democracia.

Por fim, o SEA GEO AMAZÔNICO contribui para que políticas públicas sejam elaboradas com uma visão de longo prazo. Ao invés de medidas paliativas, os gestores podem desenvolver estratégias sustentáveis e duradouras, promovendo avanços estruturais na qualidade de vida da população amazônica.

6.4 Aplicação Em Pesquisas Acadêmicas E Científicas

A importância do SEA GEO AMAZÔNICO também se reflete no campo acadêmico, onde os dados contidos no documento podem servir de base para diversas pesquisas científicas. A presença de informações organizadas permite que pesquisadores desenvolvam estudos mais aprofundados sobre a realidade socioeconômica e ambiental da Amazônia.

A análise de dados históricos possibilita a realização de pesquisas que buscam entender padrões e tendências ao longo do tempo. Essas pesquisas são fundamentais para a construção do conhecimento sobre o desenvolvimento regional e para a formulação de estratégias inovadoras para solucionar problemas recorrentes.

Além disso, os dados presentes no documento podem ser utilizados em modelagens estatísticas avançadas. Técnicas como regressão, análise de séries temporais e aprendizado de máquina podem ser aplicadas para prever cenários futuros e identificar fatores determinantes para determinadas ocorrências.

A presença de dados confiáveis e sistematizados também facilita colaborações interdisciplinares entre diferentes áreas do conhecimento. Pesquisadores de economia, geografia, biologia, sociologia e outras ciências podem utilizar as informações do SEA GEO AMAZÔNICO para desenvolver estudos conjuntos e mais completos.

Outro benefício para a academia é que os dados podem ser utilizados para a formulação de projetos de intervenção social. Universidades e centros de pesquisa podem utilizar as informações para propor soluções concretas que beneficiem diretamente as populações amazônicas.

Por fim, a divulgação de estudos baseados no SEA GEO AMAZÔNICO contribui para ampliar a compreensão global sobre a Amazônia. O compartilhamento dessas pesquisas com a comunidade científica internacional fortalece parcerias e fomenta discussões sobre a preservação e o desenvolvimento sustentável da região.

6.5 Aplicação Em Pesquisas Acadêmicas E Científicas: Estudo De Caso Em Quintais Produtivos Na Ilha De Caratateua, Belém/Pa

O artigo em questão explora a sustentabilidade socioeconômica e ambiental dos Quintais Produtivos (QP) na Ilha de Caratateua, localizada em Belém, no estado do Pará. Os Quintais Produtivos desempenham um papel cada vez mais relevante no contexto da agricultura familiar, especialmente em regiões como a Ilha de Caratateua, onde a integração de abordagens teóricas e metodológicas inovadoras tem proporcionado avanços significativos. A utilização de ferramentas como os Sistemas de Informação Geográfica (SIG) e a Análise

Regressiva Socioeconômica e Ambiental (SEA) tem sido fundamental para avaliar e promover a sustentabilidade dessas práticas agrícolas. A pesquisa focou em analisar dez atributos-chave, incluindo aspectos como renda, segurança alimentar, educação, acesso a mercados, participação comunitária, empoderamento feminino, conservação da biodiversidade, práticas de manejo sustentável e impacto ambiental.

O objetivo principal do estudo foi analisar os impactos socioeconômicos e ambientais das práticas agrícolas em Quintais Produtivos na Ilha de Caratateua, utilizando SIG e SEA como ferramentas para essa análise. Além disso, buscou-se propor estratégias de intervenção para promover a sustentabilidade e o bem-estar das famílias locais. A metodologia aplicada envolveu o mapeamento detalhado dos QPs por meio de SIG, seguido pela análise dos impactos socioeconômicos e ambientais com o auxílio do SEA. Foram coletados dados qualitativos e quantitativos por meio de pesquisa de campo e revisão bibliográfica, com o foco na avaliação de dez atributos essenciais relacionados à sustentabilidade dos QPs. A combinação desses métodos permitiu uma compreensão aprofundada dos fatores que influenciam a sustentabilidade dos Quintais Produtivos e as possíveis áreas de intervenção.

Os resultados obtidos durante a pesquisa indicaram um aumento significativo no Índice Socioeconômico e Ambiental dos Quintais Produtivos, que subiu de 0,63 para 0,93 entre 2019 e 2024. Este avanço reflete melhorias substanciais em diversos aspectos analisados, como aumento da renda das famílias, maior segurança alimentar, melhorias no acesso à educação e aos mercados, além de um aumento na participação comunitária e no empoderamento feminino. Também houve avanços na conservação da biodiversidade e na adoção de práticas de manejo sustentável, com uma redução no impacto ambiental das atividades. No entanto, o estudo também identificou desafios persistentes, como a falta de acesso a crédito e à assistência técnica, que limitam o potencial de crescimento e desenvolvimento dos QPs. Como resultado, foram propostas estratégias de intervenção para superar essas limitações e continuar avançando na sustentabilidade dos Quintais Produtivos.

A conclusão do estudo reafirma a importância dos Quintais Produtivos no fortalecimento da segurança alimentar, na geração de renda e na promoção da sustentabilidade ambiental. A pesquisa destaca a necessidade urgente de políticas públicas focadas em fortalecer essas práticas agrícolas, incluindo programas de capacitação, suporte à gestão dos recursos naturais e facilitação do acesso a recursos financeiros e técnicos. O fortalecimento dos Quintais Produtivos por meio de ações estratégicas e investimentos em capacitação pode melhorar significativamente a qualidade de vida das famílias envolvidas, promovendo um desenvolvimento sustentável e inclusivo na Ilha de Caratateua. A pesquisa evidencia que, com

o apoio adequado, os QPs podem continuar a desempenhar um papel crucial na melhoria das condições socioeconômicas e ambientais da região. (Figura 50). (Figura 51).

Figura 50 - Tabela de Estudo do SEA GEO no Estudo de Caso em Quintais Produtivos na Ilha de Caratateua.

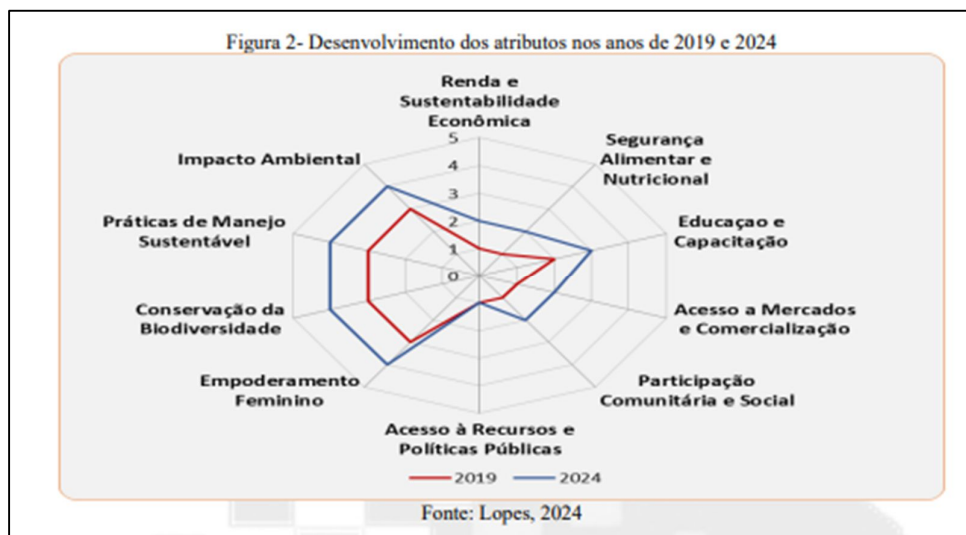
Tabela 1- Evolução do *índice socioeconômico e ambiental* nos anos de 2019 e 2024.

| Dimensão/Atributos- Escores | Ano referência | Ano Atual |
|--|----------------|-------------|
| | 2019 | 2024 |
| Renda e Sustentabilidade Econômica | 1 | 2 |
| Segurança Alimentar e Nutricional | 1 | 2 |
| Educação e Capacitação | 2 | 3 |
| Acesso a Mercados e Comercialização | 1 | 2 |
| Participação Comunitária e Social | 1 | 2 |
| Acesso à Recursos e Políticas Públicas | 1 | 1 |
| Empoderamento Feminino | 3 | 4 |
| Conservação da Biodiversidade | 3 | 4 |
| Práticas de Manejo Sustentável | 3 | 4 |
| Impacto Ambiental | 3 | 4 |
| <i>Índice Socioeconômico e Ambiental (0-1)</i> | 0,63 | 0,93 |

Fonte: Lopes, 2024.

Fonte: Desenvolvido pelo Estudo do SEA GEO no Caso em Quintais Produtivos na Ilha de Caratateua.

Figura 51 - Exemplo de Gráfico do SEA GEO no Estudo de Caso em Quintais Produtivos na Ilha de Caratateua.



Fonte: Desenvolvido pelo Estudo do SEA GEO no Caso em Quintais Produtivos na Ilha de Caratateua.

6.6 Aplicação Em Pesquisas Acadêmicas E Científicas: Análise Regressiva De Sistemas Agroecológicos: Um Estudo De Caso No Assentamento Paulo Fonteles, Distrito De Mosqueiro, Belém- Pa

O estudo em questão teve como objetivo avaliar os sistemas agroecológicos presentes no assentamento Paulo Fonteles, localizado em Belém, no estado do Pará, por meio da aplicação da metodologia SEA (Análise Regressiva Socioeconômica e Ambiental). O foco da pesquisa foi analisar a evolução desses sistemas agroecológicos, com ênfase na autonomia e capacidade de resposta dos agroecossistemas. Para isso, foram criadas duas dimensões de análise principais: autonomia, que se refere à capacidade do sistema de gerar e manter seus processos de forma independente, e capacidade de resposta, que está relacionada à flexibilidade do agroecossistema em responder a mudanças externas e internas. O estudo utilizou também geotecnologias para análise espacial, o que permitiu um mapeamento detalhado da organização dos sistemas ao longo do tempo.

A metodologia adotada envolveu a avaliação dos agroecossistemas do assentamento Paulo Fonteles entre os anos de 2006 e 2019, com o uso de escores de 1 a 5 para classificar diferentes atributos relacionados à autonomia e capacidade de resposta dos sistemas. O estudo revelou resultados importantes: a autonomia do agroecossistema apresentou um crescimento significativo de 42,6%, indicando um fortalecimento dos processos internos no assentamento. O quintal agroflorestral, em particular, teve um aumento expressivo no índice SEA, subindo 67,7%, evidenciando um avanço positivo nas práticas agroecológicas adotadas. Em contrapartida, o monocultivo de macaxeira apresentou uma diminuição de 20% no índice SEA, sugerindo que práticas mais tradicionais e menos diversificadas não apresentam os mesmos avanços em termos de sustentabilidade socioeconômica e ambiental.

Com base nos resultados obtidos, o estudo conclui que diferentes sistemas agroecológicos reagem de forma distinta a fatores internos e externos, como demonstrado pelas diferenças nos índices SEA de cada sistema. Os quintais agroflorestrais mostraram um desenvolvimento mais robusto e sustentável em comparação ao monocultivo de macaxeira, reforçando a importância da diversificação das práticas agrícolas para a promoção da sustentabilidade. Além disso, o uso de mapas temáticos se mostrou uma ferramenta valiosa na visualização e compreensão da organização espacial dos agroecossistemas, facilitando a análise das transformações e da evolução desses sistemas ao longo do tempo. O estudo

evidencia a eficácia da combinação entre a análise socioeconômica e ambiental e as geotecnologias na avaliação de práticas agroecológicas, oferecendo subsídios para a implementação de políticas públicas voltadas à promoção de sistemas agrícolas mais sustentáveis e resilientes.

Figura 52 - Tabela do SEA GEO na Análise Regressiva de Sistemas Agroecológicos.

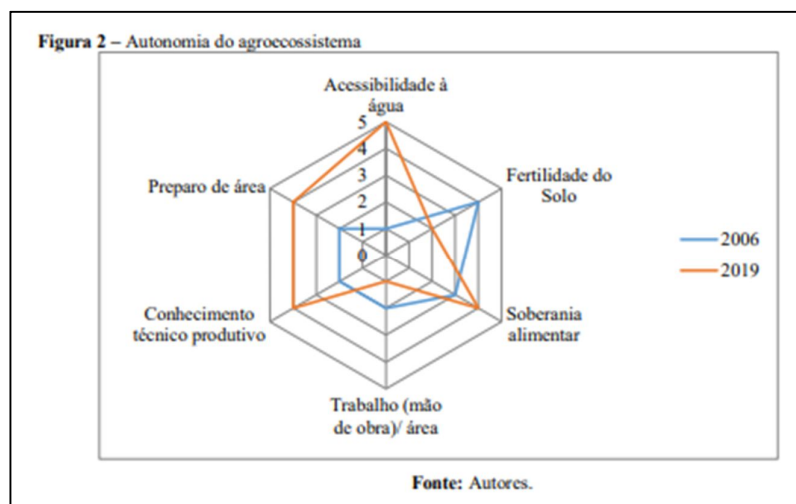
| Atributos | 2006* | 2019** |
|---|--------------|---------------|
| Acessibilidade à água | 1,0 | 5,0 |
| Fertilidade do Solo | 4,0 | 2,0 |
| Soberania alimentar | 3,0 | 4,0 |
| Trabalho (mão de obra)/ área | 2,0 | 1,0 |
| Conhecimento técnico produtivo | 2,0 | 4,0 |
| Preparo de área | 2,0 | 4,0 |
| Índice de autonomia do agroecossistema | 0,47 | 0,67 |

*= Ano de referência.
**= Ano atual.

Fonte: Autores.

Fonte: Desenvolvido pelo Estudo do SEA GEO na Análise Regressiva de Sistemas Agroecológicos.

Figura 53 - Gráfico do SEA GEO na Análise Regressiva de Sistemas Agroecológicos.



Fonte: Desenvolvido pelo Estudo do SEA GEO na Análise Regressiva de Sistemas Agroecológicos.

6.7 Metodologia Do Sea Geo Amazônico

A metodologia de funcionamento da tabela de análise socioeconômica e análise temporal baseia-se na coleta, organização e interpretação de dados referentes às condições sociais e econômicas de um determinado grupo ao longo do tempo. O objetivo principal é identificar padrões e tendências que possam auxiliar na tomada de decisão e formulação de políticas públicas ou estratégias de intervenção.

A tabela é estruturada em diferentes categorias, cada uma representando um aspecto específico da análise, como renda, educação, acesso a serviços públicos e outras variáveis socioeconômicas relevantes. Os dados são organizados em colunas e linhas, onde cada linha corresponde a um período ou grupo analisado e cada coluna representa um critério específico de avaliação.

As fórmulas utilizadas na tabela são responsáveis por realizar cálculos matemáticos e estatísticos, garantindo a precisão das análises. Entre as fórmulas mais comuns, destacam-se médias aritméticas, desvio-padrão, coeficiente de variação e cálculo de tendências baseadas em regressão linear. Essas fórmulas são essenciais para compreender o comportamento dos dados ao longo do tempo.

O funcionamento da tabela envolve a inserção dos dados coletados, que podem ser obtidos por meio de pesquisas, censos ou registros administrativos. Uma vez inseridos, os cálculos automáticos permitem identificar diferenças significativas entre os períodos analisados, evidenciando possíveis mudanças estruturais ou impactos de políticas implementadas.

Dentre as fórmulas utilizadas na tabela, destacam-se os índices de integração e recomendação, calculados por meio da seguinte expressão: $\text{*=SOMA}(B43:B47)/25$ para o ano de referência e $\text{*=SOMA}(C43:C47)/25$ para o ano atual. Essas fórmulas somam os valores contidos em um determinado intervalo de células e multiplicam o resultado por um fator fixo de 1/25. Esse coeficiente parece estar relacionado à normalização dos dados dentro de uma escala padrão, facilitando a comparação entre diferentes períodos.

A aplicação dessas fórmulas permite avaliar como os indicadores socioeconômicos evoluíram ao longo do tempo. Se o índice de integração e recomendação para o ano atual for superior ao do ano de referência, isso pode indicar melhorias na integração social e na efetividade das políticas adotadas. Caso contrário, um valor inferior pode sugerir uma piora nos indicadores, demandando novas ações ou ajustes nas estratégias implementadas.

O fator de multiplicação $1/25$ na fórmula sugere que os valores obtidos precisam ser ajustados para caber dentro de um intervalo compreensível, geralmente entre 0 e 1. Esse tipo de normalização é comum em índices socioeconômicos, pois permite que diferentes dimensões sejam comparadas de maneira uniforme, independentemente das unidades de medida originais. Isso facilita a visualização dos dados e a extração de insights para tomadas de decisão baseadas em evidências.

O gráfico gerado a partir da tabela tem a função de representar visualmente as tendências identificadas, facilitando a compreensão e interpretação dos resultados. Gráficos de linha são frequentemente utilizados para demonstrar a evolução de indicadores ao longo do tempo, enquanto gráficos de barras podem ser aplicados para comparação entre grupos distintos. A análise desses elementos é fundamental para o entendimento das variações nos indicadores e a formulação de estratégias mais eficazes.

7 CONCLUSÕES

A presente pesquisa sintetiza a reflexão sobre a automação no geoprocessamento, destacando a superação dos conflitos conceituais e das contradições identificadas durante a análise do problema. O uso do QGIS e da linguagem Python mostrou-se essencial para a otimização de processos, permitindo a conversão eficiente de dados geoespaciais para o sistema SIRGAS 2000/UTM, além da realização precisa de cálculos de áreas e perímetros e da padronização na criação de layouts cartográficos. Esses avanços evidenciam a importância de tornar as análises espaciais mais ágeis e acessíveis para diferentes aplicações, sejam acadêmicas ou profissionais.

Ao longo do estudo, foi possível constatar que a automação reduz significativamente o tempo de execução de tarefas repetitivas, minimiza erros manuais e garante maior consistência nos resultados. A superação dessas dificuldades reforça a necessidade de ferramentas que otimizem a manipulação de grandes volumes de dados espaciais, permitindo uma interpretação mais clara e eficiente das informações. A utilização de scripts em Python no ambiente do QGIS demonstrou ser uma abordagem prática e adaptável, proporcionando uma solução escalável para diferentes usuários e necessidades.

Além dos benefícios técnicos, os resultados destacam o impacto positivo dessas ferramentas no planejamento urbano, na gestão de recursos naturais e no monitoramento ambiental. A capacidade de análise e visualização dinâmica de dados espaciais contribui diretamente para uma tomada de decisão mais embasada em projetos territoriais. A padronização de layouts cartográficos, por sua vez, facilita a comunicação dos resultados, garantindo que mapas e relatórios sejam compreensíveis e úteis para diferentes públicos, desde gestores até comunidades interessadas.

Outro aspecto relevante foi a verificação da funcionalidade e precisão das ferramentas desenvolvidas. O processo de testes no QGIS confirmou que a metodologia implementada é eficaz, assegurando a confiabilidade dos resultados gerados. A automação na criação de mapas e relatórios mostrou-se eficiente, reduzindo a necessidade de ajustes manuais e proporcionando uma estrutura mais organizada e acessível para os usuários.

Dessa forma, este estudo contribui para a modernização dos métodos de análise espacial, fornecendo um modelo replicável para outras iniciativas na área. A aplicação do código desenvolvido pode ser expandida para diversas finalidades, como planejamento de infraestrutura, estudos ambientais e monitoramento territorial. Além disso, a flexibilidade do

script permite sua adaptação para diferentes contextos, ampliando suas possibilidades de uso e aumentando a eficiência de projetos futuros.

Por fim, conclui-se que a automação no geoprocessamento representa um avanço essencial para aprimorar a eficiência e qualidade das análises espaciais. O desenvolvimento de scripts em Python integrados ao QGIS otimiza processos, reduz custos operacionais e assegura maior confiabilidade nos resultados. A relevância desse trabalho transcende o contexto acadêmico, podendo impactar positivamente a gestão de dados espaciais e suas aplicações no mundo real. Espera-se que esta pesquisa inspire novas iniciativas que continuem a explorar o potencial da automação na área de geoprocessamento.

8 REFERÊNCIAS BIBLIOGRÁFICAS

BRASIL. Ministério da Saúde. Atenção Primária à Saúde no Brasil. Brasília: Ministério da Saúde, 2020. Disponível em: <https://www.gov.br/saude>. Acesso em: 15 fev. 2025.

CONFERÊNCIA NACIONAL DOS BISPOS DO BRASIL (CNBB). A Igreja e a Transformação Social no Brasil. Brasília: CNBB, 2018. Disponível em: <https://www.cnbb.org.br>. Acesso em: 15 fev. 2025.

FUNDAÇÃO GETULIO VARGAS (FGV). Déficit Habitacional no Brasil. São Paulo: FGV, 2020. Disponível em: <https://www.fgv.br>. Acesso em: 15 fev. 2025.

INSTITUTO DE PESQUISA ECONÔMICA APLICADA (IPEA). Cidades Sustentáveis e Qualidade de Vida. Brasília: IPEA, 2019. Disponível em: <https://www.ipea.gov.br>. Acesso em: 15 fev. 2025.

INSTITUTO NACIONAL DE ESTUDOS E PESQUISAS EDUCACIONAIS (INEP). Relatório Anual da Educação Básica no Brasil. Brasília: INEP, 2021. Disponível em: <https://www.gov.br/inep>. Acesso em: 15 fev. 2025.

SERVIÇO BRASILEIRO DE APOIO ÀS MICRO E PEQUENAS EMPRESAS (SEBRAE). O Impacto do Comércio no Desenvolvimento Regional. Brasília: SEBRAE, 2022. Disponível em: <https://www.sebrae.com.br>. Acesso em: 15 fev. 2025.

SANTOS, Herbert Lincon Rodrigues Alves dos. Automação Completa no QGIS. Udemy, 2022. Disponível em: <https://www.udemy.com/course/automacao-completa-no-qgis/>. Acesso em: 16 fev. 2025.

AWARI. Python: A Linguagem de Programação Essencial para o QGIS. Awari, 2023. Disponível em: <https://awari.com.br/python-a-linguagem-de-programacao-essencial-para-o-qgis/>. Acesso em: 16 fev. 2025.

COUTINHO, Lucas Amoêdo de Victor; RIBEIRO, Vitor Nere Araújo. Enpyre: uma plataforma web interativa e lúdica para o ensino de Python. 2023. Disponível em: <https://www.bdm.unb.br/handle/10483/39155>. Acesso em: 16 fev. 2025.

ROQUE, Matheus Maia. Inserção de lógica de programação no ensino básico usando linguagem Python e Biblioteca Pygame. 2017. Disponível em: <https://repositorio.ufc.br/handle/riufc/35030>. Acesso em: 16 fev. 2025.

TEIXEIRA, Robson dos Santos. Desenvolvimento de um software educacional em Python para estudo de matemática básica no ensino médio. 2022. Disponível em: <https://repositorio.uninter.com/handle/1/1314>. Acesso em: 16 fev. 2025.

Silva, A. P. V.; Vasconcelos, M. A. M.; Souza, A. P. V. de; Soares, R. S.; Bulhões, C. C. C.; Santos, A. V. F.; Bittencourt, P. C. S.; Silva, O. G. SUSTENTABILIDADE SOCIOECONÔMICA E AMBIENTAL EM QUINTAIS PRODUTIVOS (QP) NA ILHA DE CARATATEUA, BELÉM/PA: DESAFIOS E OPORTUNIDADES. *Journal of Media Critiques, Brazil*, v. 10, n. 26, p. 01-20, 2024.

ANEXO 1



ANEXO 2



ANEXO 3



ANEXO 4



ANEXO 5



ANEXO 6

