

Universidade Federal de Itajubá (UNIFEI)

Instituto de Recursos Naturais

# **Clima-Cast-Crepaldi**

## **Manual do Usuário**

Autor: Paulo César Crepaldi

Docente: Prof. Dr. Enrique Vieira Mattos

Itajubá (MG) 2025

# Sumário

<b>1</b>	<b>Apresentação</b>	<b>3</b>
1.1	O que é o Clima-Cast-Crepaldi (CCC)	3
1.2	Propósito e objetivos	3
1.3	Público-alvo	3
1.4	Escopo do sistema	3
1.5	Limitações e premissas	3
1.6	Como usar este manual	4
1.7	Conceitos Fundamentais	4
<b>2</b>	<b>Requisitos</b>	<b>4</b>
2.1	Requisitos de hardware e software	4
2.2	Requisitos de conta e autenticação	5
2.3	Dependências principais de software	5
2.4	Dependências completas de software	5
2.5	Requisitos de dados e compatibilidade	6
2.6	Requisitos de segurança e privacidade	7
2.7	Verificação de pré-requisitos	7
<b>3</b>	<b>Instalação e Acesso</b>	<b>8</b>
3.1	Acesso via Web (recomendado)	8
3.2	Execução local (opcional)	8
3.3	Atualização do aplicativo	9
3.4	Verificação de instalação e desempenho	9
<b>4</b>	<b>Fluxo de Uso Geral</b>	<b>10</b>
4.1	Etapas principais de interação	10
4.2	Diagrama geral do fluxo de uso	11
4.3	Considerações sobre o fluxo	11
<b>5</b>	<b>Interface do Usuário</b>	<b>12</b>
5.1	Visão geral da interface	12
5.2	Barra lateral (controles de entrada)	12
5.2.1	Detalhamento dos modos interativos	14
5.3	Área central (resultados e visualização)	15
5.3.1	Ferramentas de análise gráfica	18
5.4	Ferramentas de exportação	19
<b>6</b>	<b>Resolução de Problemas e Diagnóstico</b>	<b>19</b>
6.1	Mensagens de erro comuns	19

6.2	Verificação de autenticação GEE . . . . .	20
6.3	Diagnóstico de desempenho . . . . .	21
6.4	Verificação de ambiente Python . . . . .	21
6.5	<b>Limpeza de cache do Streamlit.</b> . . . .	21
6.6	Recuperação do ambiente . . . . .	22
6.7	Relato de bugs . . . . .	22
<b>Apêndice Técnico</b>		<b>23</b>
B.	Variáveis meteorológicas disponíveis (ERA5-Land) . . . . .	24
C.	Estrutura do repositório GitHub . . . . .	25
D.	Contatos e suporte . . . . .	25

# 1. Apresentação

## 1.1. O que é o Clima-Cast-Crepaldi (CCC)

O Clima-Cast-Crepaldi (CCC) é um aplicativo interativo para análise de variáveis meteorológicas, derivadas de reanálises, desenvolvido no contexto da disciplina *Ferramentas de Previsão de Curtíssimo Prazo (Nowcasting)* — CAT314, do curso de Ciências Atmosféricas da Universidade Federal de Itajubá (UNIFEI).

O sistema integra o *Google Earth Engine* (GEE) para processamento geoespacial em nuvem e utiliza o conjunto de dados **ERA5-Land**, cuja resolução espacial típica é de aproximadamente 9 km. O aplicativo gera mapas (interativos e estáticos) e séries temporais a partir dos produtos disponibilizados (*datasets*), otimizando, assim, o seu desempenho.

## 1.2. Propósito e objetivos

- Fornecer visualização cartográfica com paletas discretas e intervalos padronizados para variáveis meteorológicas;
- Disponibilizar séries temporais com estatísticas resumidas (valores máximo e mínimo, média, Amplitude e Desvio Padrão) para áreas de interesse;
- Permitir exportação direta de mapas (JPG e PNG) e gráficos em formatos XLSX e CSV.

## 1.3. Público-alvo

- Discentes de Ciências Atmosféricas e áreas afins;
- Docentes e pesquisadores interessados em diagnósticos rápidos com base em reanálises;
- Profissionais que necessitam de análises espaciais e temporais regionais.

## 1.4. Escopo do sistema

- **Dados:** ERA5-Land e seus agregados mensais, diários e horários, priorizando variáveis de superfície, como precipitação total, temperatura média a 2 m, temperatura do solo, vento a 10 m, radiação solar incidente, umidade relativa derivada e umidade do solo (quatro profundidades);
- **Processamento:** seleção de áreas de interesse (estado, município, círculo e polígono), intervalo temporal (anual, mensal e personalizado) e renderização via GEE;
- **Produtos:** mapas com legenda discreta e séries temporais agregadas diariamente.

## 1.5. Limitações e premissas

- As reanálises não substituem observações *in situ*;
- A resolução de ~9 km e a agregação diária suavizam extremos horários;
- O desempenho depende da estabilidade da rede e das políticas de uso do GEE;
- Existe uma única opção de dado horário para não comprometer o tempo de processamento.

## 1.6. Como usar este manual

O presente manual foi elaborado de modo a adotar uma estrutura modular e progressiva:

1. **Visão geral** — introdução conceitual e fluxos principais;
2. **Procedimentos passo a passo** — tarefas típicas com validação e saída esperada;
3. **Referências Internas** — listas de variáveis e parâmetros do sistema;
4. **Resolução de problemas** — diagnóstico de falhas e boas práticas de desempenho.

## 1.7. Conceitos Fundamentais

Para facilitar a compreensão do funcionamento CCC, esta seção apresenta brevemente três conceitos essenciais utilizados ao longo do manual. **Reanálises atmosféricas.** Reanálises são conjuntos de dados globais gerados a partir da combinação entre observações meteorológicas (estações em superfície, radiossondagens, satélites) e modelos numéricos. Esse processo integra todas essas informações e produz séries temporais, espacialmente completas e fisicamente consistentes. O CCC utiliza a reanálise ERA5-Land produzida pelo *European Centre for Medium-Range Weather Forecasts* (ECMWF).

**ERA5-Land.** O ERA5-Land é um produto derivado do ERA5 com maior detalhamento para variáveis de superfície. Possui resolução aproximada de 9 km e fornece temperatura, vento, precipitação, radiação e outras variáveis relevantes para análises ambientais e climáticas.

**Google Earth Engine (GEE).** O GEE é uma plataforma de computação em nuvem voltada para processamento geoespacial em larga escala. Ela permite acessar diretamente *datasets* de dados de reanálises, imagens de satélite e modelos climáticos, além de executar cálculos sobre áreas definidas pelo usuário. O CCC utiliza o GEE para realizar recortes espaciais, agregações e renderização de mapas.

## 2. Requisitos

Esta seção descreve os **pré-requisitos técnicos e operacionais** necessários para o funcionamento correto do sistema, bem como os ambientes suportados, bibliotecas, conexões e dados disponíveis.

### 2.1. Requisitos de hardware e software

O CCC foi concebido como uma aplicação *web-based* desenvolvida em Python e Streamlit, acessível via navegador. Alguns parâmetros devem ser observados:

- **Navegador compatível:** Google Chrome, Mozilla Firefox ou Microsoft Edge em suas versões mais atuais;
- **Resolução recomendada:** mínima de  $1366 \times 768$  px; ideal de  $1920 \times 1080$  px;
- **Memória RAM:** superior a 4 GB para execução fluida e cache local de figuras;

- **Sistema operacional:** Windows 10 ou 11, macOS 12 ou superior, ou distribuições Linux (64 bits);
- **Conexão de internet:** estável, superior a 5 Mbps; latência inferior a 200 ms para evitar atraso nos mapas.

Esses parâmetros asseguram que a interface Streamlit e as respostas do GEE sejam renderizadas em tempo aceitável.

## 2.2. Requisitos de conta e autenticação

A plataforma foi desenvolvida para suportar dois modelos de autenticação distintos, garantindo a portabilidade entre o ambiente de desenvolvimento local e o deployment em nuvem:

1. Usuário autenticado por **conta Google** com acesso à API Earth Engine;
2. Em execução local, é necessário o **setup inicial** via `ee.Authenticate()` (que salva as credenciais) seguido, em tempo de execução, apenas de `ee.Initialize()`;
3. Em ambiente *Streamlit Cloud*, o acesso é feito via credenciais de *service account* registradas em `st.secrets`;
4. Permissão de leitura do dataset ECMWF/ERA5\_LAND\_DAILY\_AGGR.

O aplicativo detecta automaticamente se a autenticação está ativa e exibe mensagens de status (Conectado ao GEE ou Falha na autenticação).

## 2.3. Dependências principais de software

O ambiente Python deve conter as seguintes bibliotecas (versões mínimas recomendadas):

## 2.4. Dependências completas de software

O ambiente Python utilizado segue o arquivo `requirements.txt` oficial do projeto, garantindo compatibilidade com o *Google Earth Engine* (GEE) e o ambiente *Streamlit Cloud*. A Tabela 1 apresenta todas as dependências e suas versões mínimas recomendadas.

Tabela 1: Bibliotecas e versões mínimas recomendadas para o CCC

Biblioteca	Versão mínima
numpy	1.26.4
matplotlib	3.8.4
streamlit	1.45.1
folium	0.17.0
pandas	2.3.3
plotly	5.23.0
geopandas	1.0.0
earthengine-api	1.5.11
geemap	0.35.3
streamlit_folium	0.22.0
setuptools	75.1.0
geobr	0.2.2
openpyxl	3.1.5
pyproj	3.6.1
python-docx	1.1.2

Essas bibliotecas cobrem:

- **Visualização e interface:** streamlit, streamlit\_folium, plotly;
- **Mapeamento e geoprocessamento:** geemap, folium, geopandas, geobr, pyproj;
- **Manipulação de dados:** pandas, numpy, openpyxl;
- **Documentação e relatórios:** python-docx;
- **Infraestrutura e compatibilidade:** earthengine-api, setuptools.

O uso das versões especificadas garante estabilidade em ambientes Python 3.12 (Streamlit Cloud) e Python 3.13 (execução local em Windows/Linux). A atualização para versões superiores deve ser feita com cautela, verificando a compatibilidade entre geemap e earthengine-api, que compartilham dependências internas.

Essas dependências garantem compatibilidade total com o GEE Python API e suporte a gráficos interativos. A versão consolidada é mantida no arquivo `requirements.txt` do repositório GitHub.

## 2.5. Requisitos de dados e compatibilidade

- **Fonte principal:** ERA5-Land (Daily Aggregated);
- **Cobertura temporal:** 1950 – presente;
- **Resolução espacial:**  $\approx 9$  km ( $0.1^\circ$ );
- **Unidades padrão:** precipitação (mm), temperatura ( $^\circ\text{C}$ ), vento (m/s), umidade relativa (%) e radiação solar ( $\text{W}/\text{m}^2$ );
- **Sistemas de referência:** coordenadas WGS 84 (EPSG 4326).

Os limites administrativos utilizados nos mapas são obtidos via pacote `geobr` (IBGE), garantindo compatibilidade com polígonos municipais e estaduais do Brasil.

## 2.6. Requisitos de segurança e privacidade

Nenhum dado pessoal é coletado pelo aplicativo. As comunicações com o GEE e o Streamlit Cloud são realizadas via HTTPS. Caso o usuário autentique com sua conta Google, o token de sessão é armazenado localmente e nunca compartilhado.

## 2.7. Verificação de pré-requisitos

O sistema executa uma rotina automática ao ser iniciado:

1. Testa a conectividade com o GEE (`ee.Image.constant(0).getInfo()`);
2. Verifica versões das bibliotecas principais;
3. Confirma o acesso ao dataset ERA5-Land;
4. Exibe na interface uma mensagem de status da conexão.

Essa verificação impede falhas de execução e fornece diagnóstico rápido do ambiente.



### 3. Instalação e Acesso

Esta seção descreve os procedimentos de instalação e acesso ao CCC, abrangendo o uso direto via Web — principal forma de acesso — e as instruções resumidas de execução local apenas para fins de desenvolvimento.

#### 3.1. Acesso via Web (recomendado)

- Acesse o aplicativo diretamente no navegador por meio do endereço:  
<https://climacastcrepaldiv2.streamlit.app>
- Após o carregamento da página, o aplicativo realiza automaticamente o teste de autenticação com o Google Earth Engine (GEE);
- Utilize a barra lateral para selecionar:
  1. A área de interesse (Estado, Município, Polígono ou Círculo);
  2. O período de análise;
  3. A variável meteorológica (ex.: precipitação, temperatura, umidade relativa);
  4. O tipo de produto desejado: mapa ou série temporal.

#### Vantagens:

- Não requer instalação local;
- Atualizações automáticas de código e bibliotecas;
- Processamento em nuvem (Google Earth Engine + Streamlit Cloud);
- Compatibilidade total com Windows, macOS e Linux.

#### Requisitos:

- Conexão à internet estável (superior a 5 Mbps);
- Navegador atualizado (Chrome, Edge, Firefox);
- Permitir execução de scripts JavaScript.

#### Limitações:

- O tempo de resposta depende da área selecionada e do tamanho do período solicitado;
- O Streamlit Cloud impõe limite de execução ( 60 segundos por processo).

#### 3.2. Execução local (opcional)

Usuários avançados podem executar o CCC localmente para fins de desenvolvimento ou integração com outras aplicações Python. As etapas resumidas são:

1. Clonar o repositório oficial:

```
git clone https://github.com/Crepaldi2025/dashboard_cat314.git
cd dashboard_cat314
```

2. Instalar dependências:

```
pip install -r requirements.txt
```

3. Autenticar no Google Earth Engine:

```
earthengine authenticate
```

4. Executar o aplicativo:

```
streamlit run main.py
```

O sistema abrirá automaticamente em <http://localhost:8501>.

### 3.3. Atualização do aplicativo

- Para a versão Web, as atualizações são automáticas;
- Para execução local, atualize manualmente com:

```
git pull origin main  
pip install -U -r requirements.txt
```

### 3.4. Verificação de instalação e desempenho

Recomenda-se testar a execução com uma área simples:

- Selecione o estado de Minas Gerais;
- Defina o período 2024-01-01 a 2024-01-10;
- Escolha a variável Precipitação total (mm).

O mapa deverá ser renderizado com paleta discreta de azuis e escala 0–100 mm. Se não houver resposta em até 20 s, verifique conectividade e autenticação GEE.

## 4. Fluxo de Uso Geral

Esta seção apresenta a sequência lógica de ações realizadas pelo usuário dentro do sistema, desde o acesso inicial até a exportação dos produtos gerados. O objetivo é fornecer uma visão global do funcionamento do aplicativo e facilitar o entendimento de sua estrutura modular.

**Nota sobre desempenho:** O tempo de resposta do sistema pode variar conforme a latência da rede do usuário e a demanda momentânea nos servidores do Google Earth Engine. Áreas muito extensas ou intervalos temporais longos também podem aumentar o tempo de processamento.

### 4.1. Etapas principais de interação

O fluxo geral de uso pode ser dividido em seis etapas sequenciais:

1. **Acesso ao sistema** — o usuário abre o aplicativo no navegador:  
<https://climacastcrepaldiv2.streamlit.app> e aguarda a inicialização automática das bibliotecas e autenticação com o Google Earth Engine;
2. **Seleção da área de interesse** — o usuário escolhe o modo de delimitação desejado:
  - Estado brasileiro;
  - Município (via base do IBGE/`geobr`);
  - Polígono desenhado no mapa;
  - Círculo de raio definido (em km) a partir de um ponto central.
3. **Definição do período de análise** — inserção das datas inicial e final. O aplicativo limita automaticamente o intervalo conforme a disponibilidade da série ERA5-Land;
4. **Escolha da variável meteorológica** — o usuário seleciona a variável desejada, como:
  - Precipitação total (mm);
  - Temperatura média a 2 m (°C);
  - Temperatura da superfície (°C);
  - Umidade relativa (%);
  - Vento a 10 m (m/s);
  - Radiação solar incidente (W/m<sup>2</sup>),
  - Umidade do solo (4 profundidades).
5. **Geração do produto** — o sistema processa os dados via GEE e gera:
  - Mapa interativo;
  - Série temporal.
6. **Exportação de resultados** — o usuário pode salvar:
  - Mapas em formatos `.png` ou `.jpg`;
  - Séries temporais em `.csv` ou `.xlsx`.

## 4.2. Diagrama geral do fluxo de uso

A Figura 1 ilustra o processo completo de operação do CCC, desde o acesso até a exportação dos resultados.

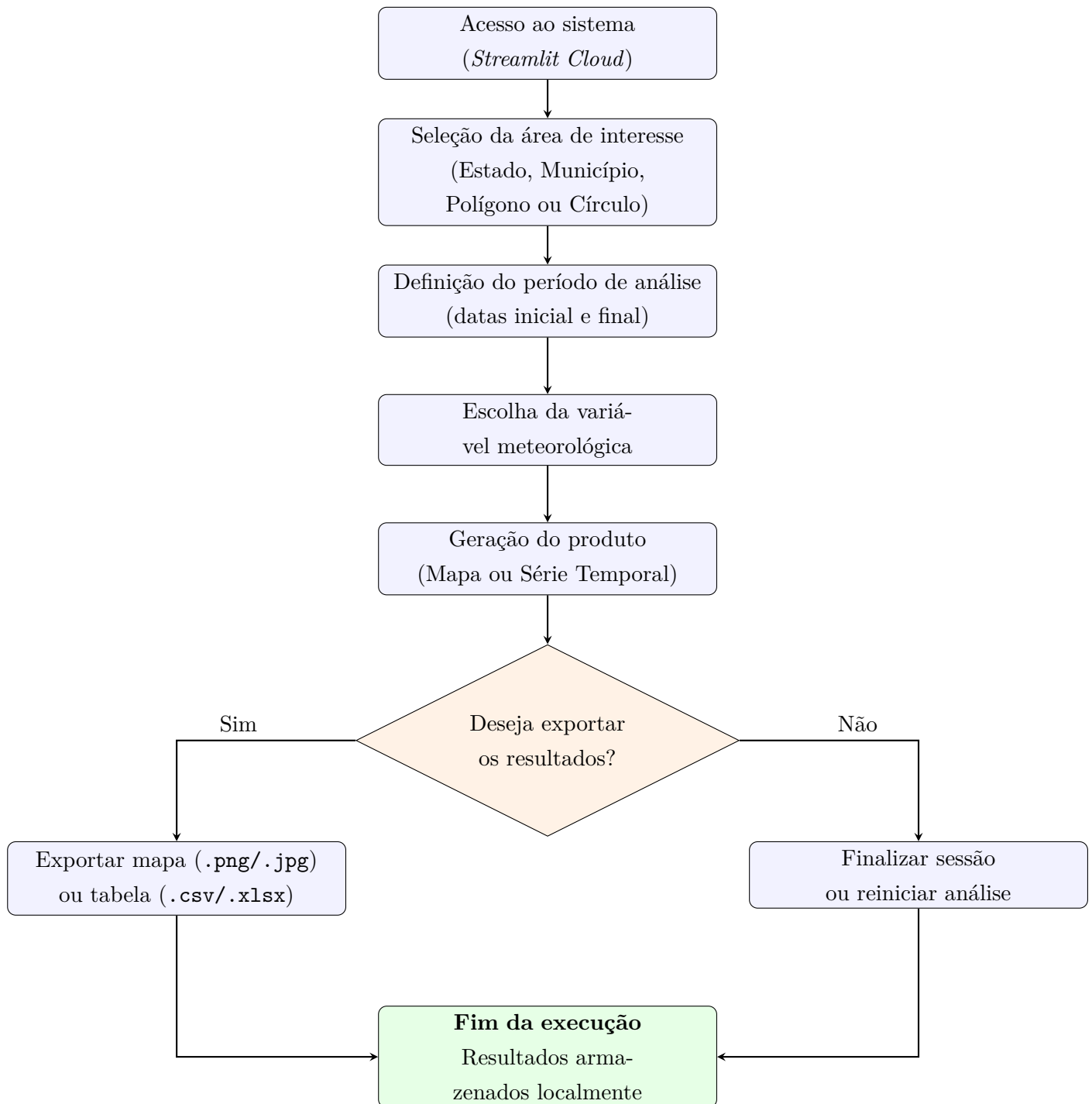


Figura 1: Fluxograma geral de utilização do Clima-Cast-Crepaldi.

## 4.3. Considerações sobre o fluxo

- O sistema é modular: o usuário pode reiniciar a análise a qualquer momento, alterando apenas um dos parâmetros (área, período ou variável);
- A renderização de mapas é otimizada pelo uso dos *datasets* ERA5- Land;

- Caso o tempo de processamento exceda o limite do servidor, o aplicativo exibirá uma mensagem de aviso com a opção de reduzir a área ou o intervalo de datas;
- Todos os resultados permanecem disponíveis na aba atual do navegador até que o usuário recarregue a página.

## 5. Interface do Usuário

A interface foi projetada para ser clara, intuitiva e responsiva, permitindo que usuários de diferentes níveis técnicos explorem os dados meteorológicos disponíveis. Esta seção descreve a estrutura visual do sistema, os componentes interativos e as funções principais disponíveis na tela.

### 5.1. Visão geral da interface

A Figura 2 apresenta o layout principal do CCC, composto por duas regiões principais:

1. **Barra lateral (sidebar)** — localizada à esquerda, reúne os controles de entrada e opções de configuração;
2. **Área central (painel principal)** — exibe os resultados, incluindo mapas, gráficos ou mensagens de status.

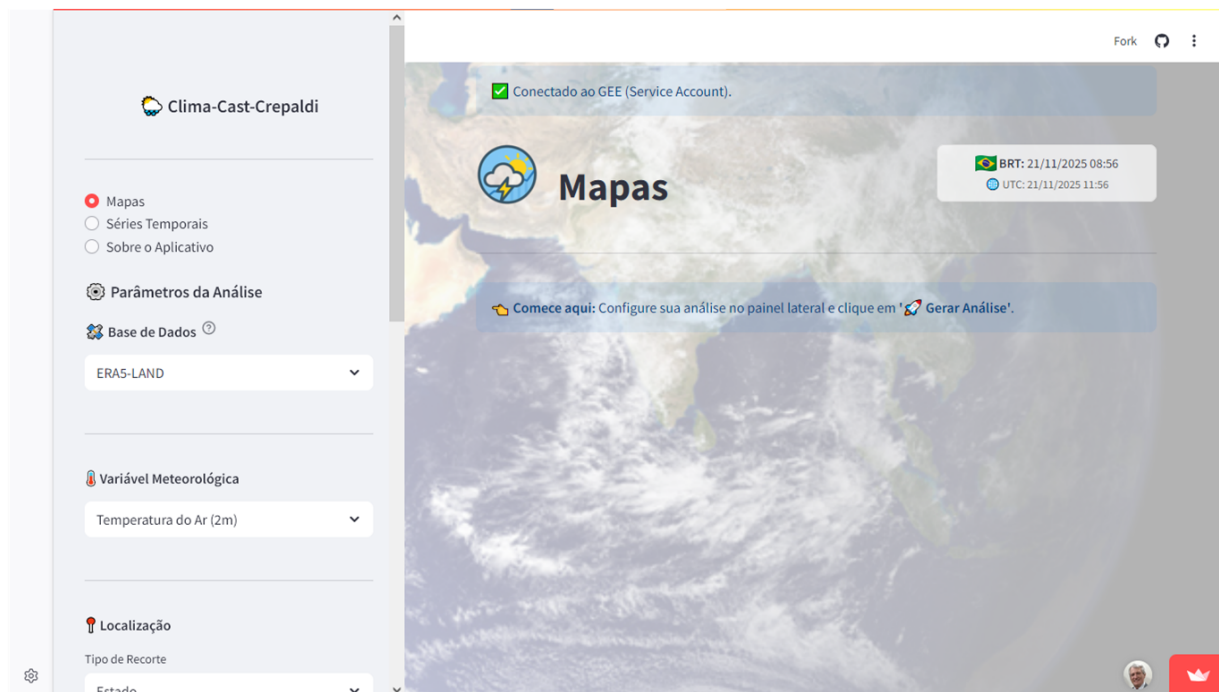


Figura 2: Layout geral da interface do aplicativo.

### 5.2. Barra lateral (controles de entrada)

A barra lateral contém os componentes de seleção que definem a consulta no Google Earth Engine. Os elementos são organizados de cima para baixo, seguindo a lógica de fluxo mostrada

na Seção 4.

### 1. Seleção de área de interesse:

- **Estado** — lista suspensa com as 27 unidades federativas;
- **Município** — campo dependente do Estado escolhido;
- **Polígono** — ferramenta de desenho livre no mapa;
- **Círculo** — coordenadas centrais e raio (em km).



Figura 3: Detalhe do menu de seleção de localização com os tipos de recorte espacial disponíveis.

### 2. Período de análise:

- Campos com datas mensais e anuais;
- Campos de data personalizada - inicial e final (YYYY-MM-DD);
- O aplicativo ajusta automaticamente para o intervalo disponível no dataset ERA5-Land.



Figura 4: Campos de definição do período de análise.

### 3. Variável meteorológica:

- Precipitação total (mm);
- Temperatura média a 2 m (°C);
- Temperatura do Ponto de Orvalho (°C);
- Temperatura da Superfície (*Skin*) (°C);
- Umidade do Solo em 4 camadas ( $m^3/m^3$ );

- Umidade relativa média (%);
- Velocidade do vento a 10 m (m/s);
- Radiação solar incidente (W/m<sup>2</sup>).

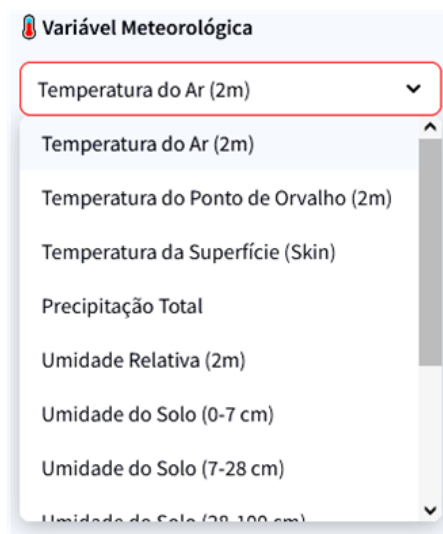


Figura 5: Menu de seleção de variáveis meteorológicas disponíveis no sistema.

#### 4. Tipo de produto:

- Mapa espacial;
- Série temporal.

#### 5. Botão de execução:

- “**Gerar Análise**” — inicia o processamento;
- Mensagens de progresso e diagnóstico aparecem na tela principal.



Figura 6: Botão de execução para iniciar o processamento e renderização dos dados.

##### 5.2.1 Detalhamento dos modos interativos

Enquanto as opções de Estado e Município utilizam limites administrativos pré-definidos, os modos de Polígono e Círculo permitem análises personalizadas.

**Uso da ferramenta Polígono** Ao selecionar este modo, utilize a barra de ferramentas localizada no próprio mapa. Selecione o ícone de desenho geométrico, clique sobre o mapa para marcar os vértices da área de interesse e clique novamente no ponto inicial para fechar a forma.



Figura 7: Exemplo de delimitação de área personalizada utilizando a ferramenta de desenho de polígono.

**Definição via Círculo** Para áreas circulares, o sistema solicita a inserção manual das coordenadas e da dimensão da área diretamente na barra lateral.

Tipo de Recorte

Círculo (Lat/Lon/Raio) ▾

Lat                      Lon

-22.4200   -   +      -45.4600   -   +

Raio (km)

10.00                      -   +

**Ajuda: Definindo o Círculo** ▴

**Como preencher:**

- **Lat/Lon:** Graus decimais (ex: -22.42).
- **Raio:** Km do centro à borda.

Figura 8: Parâmetros de entrada para área circular: coordenadas e raio (km).

### 5.3. Área central (resultados e visualização)

Dependendo da opção selecionada, a área central do aplicativo exibe um dos três tipos de resultados: **mapa interativo**, **mapa estático** ou **série temporal**. Todos são gerados a partir dos dados ERA5-Land via Google Earth Engine (GEE).

#### Mapa interativo

- Renderizado dinamicamente com **geemap** e **folium**;
- Permite zoom, deslocamento e consulta de coordenadas;
- Apresenta paleta de cores *colorbar* discreta;
- Possui título automático e indicadores de unidade;



- Ideal para exploração visual e diagnóstico rápido.

### Precipitação Total mensal (Janeiro de 2024) no estado de Minas Gerais

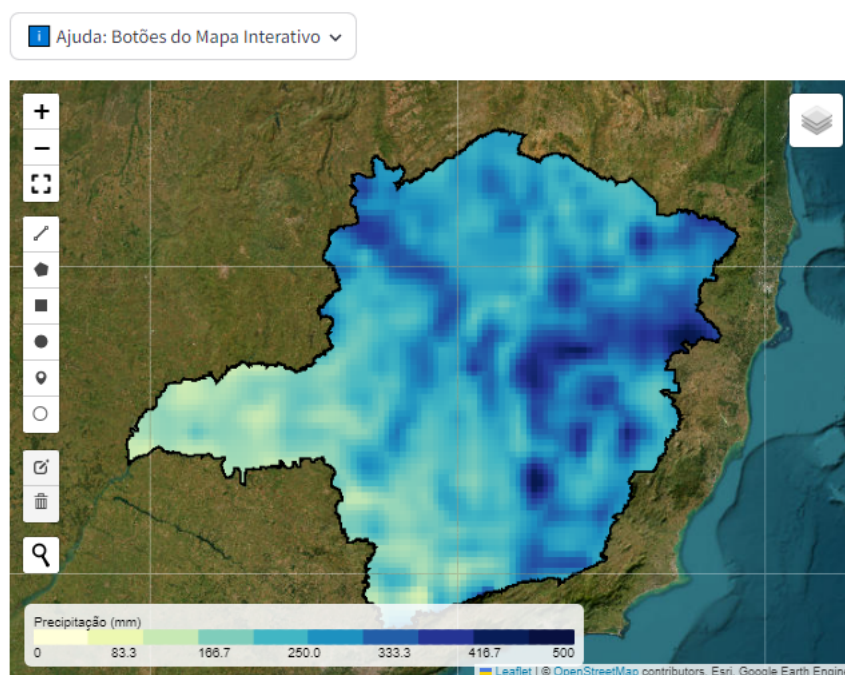


Figura 9: Exemplo de mapa interativo de precipitação (ERA5-Land, Minas Gerais, janeiro de 2024).

**Guia de ícones integrado** Para facilitar o uso das ferramentas de desenho no mapa interativo gerado, o sistema possui: Ajuda: Botões do Mapa Interativo.

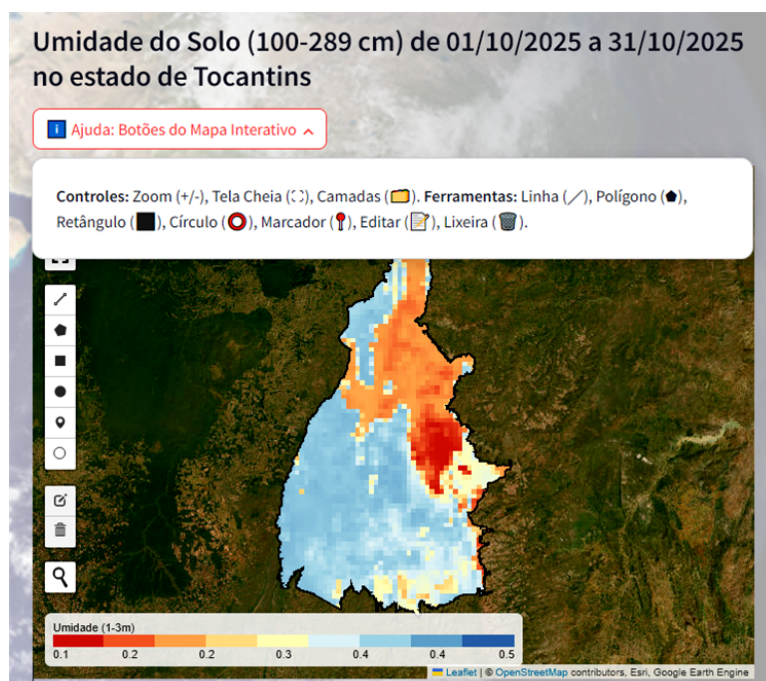


Figura 10: Menu de ajuda integrado que detalha as funções dos botões do mapa interativo.

## Mapa estático

Além do mapa interativo, o CCC gera um **mapa estático** de boa qualidade, adequado para relatórios, slides e artigos científicos. Esse mapa é salvo em formato raster (.png ou .jpg).

### Checklist antes de exportar

1. Confirmar a área e o período selecionados;
2. Verificar unidades e intervalos da colorbar;
3. Validar título e rodapé (variável, dataset, datas, fonte);
4. Escolher a resolução adequada (300 dpi ou 150 dpi).

**Temperatura do Ar (2m) de 01/07/2025 a 15/07/2025 no estado de Santa Catarina**

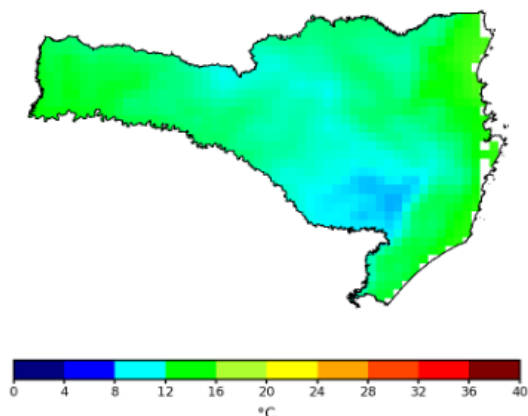


Figura 11: Exemplo de mapa estático de temperatura (ERA5-Land, Santa Catarina, 01–15 julho 2025).

## Série temporal

- Gráfico dinâmico construído com `plotly`;
- Permite interação com o cursor para leitura de valores;
- Exibe valores máximos, mínimos e médios diários além das estatísticas de Amplitude e Desvio Padrão.

### Série Temporal de Umidade Relativa (2m) de 01/12/2015 a 31/01/2016 no município de Itajubá

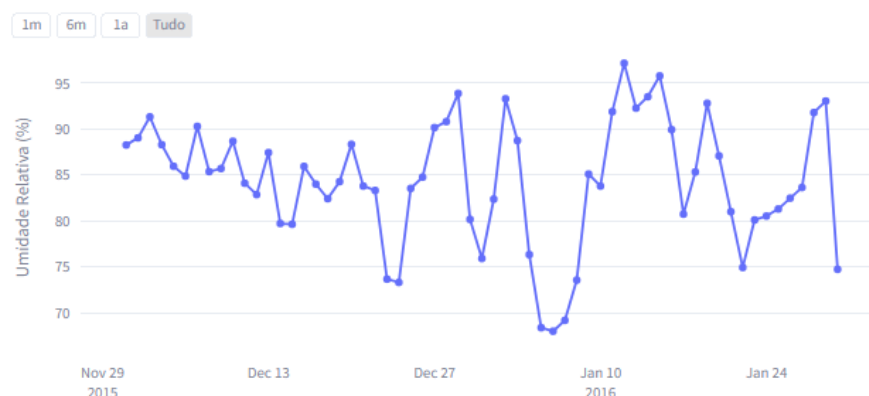


Figura 12: Exemplo de série temporal da umidade relativa para o município de Itajubá.

#### 5.3.1 Ferramentas de análise gráfica

A interface de séries temporais utiliza a biblioteca Plotly, oferecendo recursos avançados de navegação nos dados.

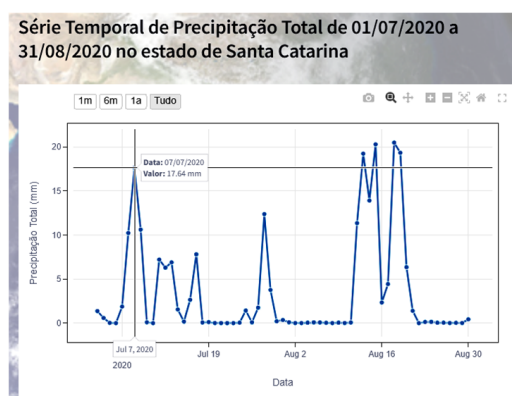


Figura 13: Exemplo de interatividade no gráfico: barra de ferramentas (canto superior direito) e leitura de valores exatos ao passar o mouse.

Para garantir que o usuário utilize todo o potencial dessas ferramentas, o sistema inclui um guia rápido acessível através do menu “Ajuda: Entenda os ícones e ferramentas do gráfico”.

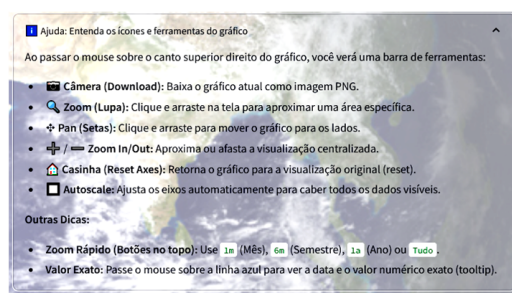


Figura 14: Painel de ajuda integrado explicando as funções de zoom, pan e download do gráfico.

## 5.4. Ferramentas de exportação

Após o processamento e visualização dos dados, o CCC disponibiliza botões dedicados para salvar os resultados localmente. Estes controles situam-se, geralmente, logo abaixo ou acima da área de visualização principal.

**Exportação de Mapas:** Para os mapas estáticos, o sistema gera arquivos de imagem rasterizada. O usuário deve utilizar o botão de download específico para salvar a figura gerada.

**Exportação de Dados:** Para dados tabulares, o sistema permite o download da série histórica completa processada. Estão disponíveis dois formatos principais:

.CSV (Comma-Separated Values): Arquivo para importação em softwares de programação ou análise estatística;

.XLSX (Excel): Planilha formatada compatível com Microsoft Excel e LibreOffice Calc.

# 6. Resolução de Problemas e Diagnóstico

Esta seção fornece orientações para diagnóstico e correção de problemas durante o uso do CCC. As situações descritas foram identificadas durante o desenvolvimento e validação do sistema, abrangendo: autenticação do GEE, erros de dependências, lentidão de execução e falhas na renderização de mapas.

Atenção a falhas de rede: O carregamento do mapa ou da série temporal pode falhar caso a conexão do usuário esteja instável ou apresente alta latência. Mensagens como “*Connecting...*”, ou carregamento infinito, geralmente indicam problemas na comunicação HTTPS entre o navegador, o Streamlit Cloud e a API do Google Earth Engine. Recomenda-se testar outra rede, utilizar conexão cabeada ou reiniciar o roteador antes de repetir a consulta.

## 6.1. Mensagens de erro comuns

A Tabela 2 apresenta as mensagens mais frequentes e suas causas prováveis.

Tabela 2: Mensagens de erro e ações corretivas

Mensagem exibida	Ação recomendada
EEException: Authentication required	Execute <code>earthengine authenticate</code> no terminal, refaça o login Google e reinicie o aplicativo.
Permission denied: projects/...	Verifique se a <i>Service Account</i> usada no <code>st.secrets</code> possui permissão de leitura no projeto GEE.
ModuleNotFoundError: No module named 'geemap'	Reinstale dependências: <code>pip install -r requirements.txt</code> .
This app has encountered an error (Streamlit Cloud)	<b>Cloud:</b> Limpe o cache do navegador e recarregue. <b>Local:</b> Use <code>streamlit cache clear</code> . Em ambos os casos, tente reduzir a área ou o período de análise.
Timeout: Earth Engine computation exceeded limit	Selecione área menor ou período reduzido. O GEE impõe limite de tempo por requisição.
QuotaExceededError	O limite de requisições do usuário ou da Service Account foi excedido. Aguarde alguns minutos e reduza a área ou o período da análise.
HTTP 429: TooManyRequests	Alta demanda momentânea nos servidores do Google Earth Engine. Tente novamente após 1–2 minutos ou reduza o volume da consulta.
InternalError: Earth Engine internal problem	Erro interno e temporário do servidor GEE. Não está relacionado ao código do CCC. Repita a operação após alguns instantes.

## 6.2. Verificação de autenticação GEE

1. Execute:

```
earthengine authenticate
```

2. Após o login no navegador, verifique o arquivo de credenciais:

```
~/.config/earthengine/credentials
```

3. Para testar a conexão diretamente em Python:

```
import ee
ee.Initialize()
print(ee.Image.constant(0).getInfo())
```

Se o teste retornar um dicionário válido, a conexão com o GEE está funcional.

### 6.3. Diagnóstico de desempenho

A lentidão na renderização pode estar associada a:

- Seleção de períodos muito longos (ex.: > 1 ano);
- Áreas de interesse muito extensas (ex.: todo o Brasil);
- Conexão instável ou com latência > 200 ms;
- Execução simultânea por múltiplos usuários no Streamlit Cloud.

### 6.4. Verificação de ambiente Python

Para confirmar versões instaladas:

```
python -m pip list
```

Compare com as versões mínimas recomendadas na Tabela 1. Diferenças podem causar incompatibilidade entre bibliotecas.

### 6.5. Limpeza de cache do Streamlit.

Em casos de resultados inconsistentes, mapas que não carregam ou mensagens de erro persistentes, recomenda-se realizar a limpeza do cache. O procedimento depende do ambiente de execução:

- **Execução local:** limpar o cache pelo terminal, antes de reiniciar o aplicativo:

```
streamlit cache clear
```

- **Execução no Streamlit Cloud:** utilizar o menu interno do aplicativo. No canto superior direito da interface do CCC, clique em:

"3 pontinhos" → Clear cache

Após a limpeza, recarregue a página. Esse procedimento remove estados anteriores, incluindo dados temporários, variáveis de sessão e versões antigas dos gráficos, garantindo que as bibliotecas e funções sejam carregadas corretamente na próxima execução.

## 6.6. Recuperação do ambiente

Em caso de falhas persistentes:

1. Exclua o diretório `venv/` e crie um novo ambiente virtual;
2. Reinstale dependências:

```
pip install -r requirements.txt
```

3. Teste a inicialização local com:

```
streamlit run main.py
```

Se o erro persistir, verifique logs no painel “Manage App” (Streamlit Cloud) ou consulte os arquivos de log locais em `.streamlit/`.

## 6.7. Relato de bugs

Erros não resolvidos devem ser reportados ao repositório oficial:

[https://github.com/Crepaldi2025/dashboard\\_cat314/issues](https://github.com/Crepaldi2025/dashboard_cat314/issues)

Inclua:

- Descrição do problema e data/hora;
- Sistema operacional e versão do Python;
- Mensagem de erro completa (copiada do terminal);
- Captura de tela, se possível.

## Apêndice Técnico

Este apêndice reúne informações complementares úteis para operação e manutenção do sistema, incluindo estrutura de diretórios, lista de variáveis disponíveis, metadados e convenções de nomenclatura.

### A. Estrutura de diretórios do projeto

A Tabela 3 apresenta a organização dos arquivos principais do aplicativo.

Tabela 3: Estrutura atual de diretórios e arquivos do repositório `dashboard_cat314`.

Caminho / Arquivo	Descrição
<code>.devcontainer/</code>	Configuração do ambiente de desenvolvimento no Visual Studio Code e Streamlit Cloud.
<code>charts_visualizer.py</code>	Geração de séries temporais e gráficos interativos (biblioteca Plotly).
<code>gee_handler.py</code>	Inicialização e requisições ao Google Earth Engine (ERA5-Land).
<code>logo.png</code>	Logotipo utilizado na interface principal do aplicativo.
<code>main.py</code>	Ponto de entrada do sistema ( <i>Streamlit</i> ), inicializando toda a aplicação.
<code>map_visualizer.py</code>	Geração e renderização de mapas interativos e estáticos.
<code>municipios_ibge.json</code>	Arquivo de referência geográfica contendo os limites municipais oficiais (IBGE).
<code>packages.txt</code>	Lista de dependências adicionais utilizadas no ambiente de execução (Cloud).
<code>requirements.txt</code>	Lista principal de bibliotecas Python e respectivas versões compatíveis.
<code>runtime.txt</code>	Definição da versão do interpretador Python (3.12) para o Streamlit Cloud.
<code>sobre.docx</code>	Documento descritivo sobre o projeto e sua apresentação.
<code>ui.py</code>	Interface de navegação e estrutura das páginas do aplicativo.
<code>utils.py</code>	Funções utilitárias de suporte às rotinas principais do sistema.



## B. Variáveis meteorológicas disponíveis (ERA5-Land)

O CCC utiliza o conjunto ECMWF/ERA5\_LAND\_DAILY\_AGGR, processado via GEE. A Tabela 4 descreve as variáveis acessíveis ao usuário.

Tabela 4: Variáveis ERA5-Land disponíveis no CCC.

Nome no sistema	Unidade	Descrição física	Origem temporal
Precipitação total	mm	Soma diária da taxa de precipitação (líquida e sólida).	Soma das estimativas horárias.
Temperatura média a 2 m (T2m)	°C	Temperatura média do ar a 2 metros da superfície.	Média das estimativas horárias.
Temperatura do Ponto de Orvalho	°C	Temperatura na qual o ar deve ser resfriado para saturar.	Média das estimativas horárias.
Temperatura da Superfície (Skin)	°C	Temperatura da interface entre a superfície da Terra (solo/vegetação) e a atmosfera.	Média das estimativas horárias.
Umidade do Solo (0–7 cm)	$m^3/m^3$	Conteúdo volumétrico de água na camada superficial (0 a 7 cm).	Média das estimativas horárias.
Umidade do Solo (7–28 cm)	$m^3/m^3$	Conteúdo volumétrico de água na segunda camada (7 a 28 cm).	Média das estimativas horárias.
Umidade do Solo (28–100 cm)	$m^3/m^3$	Conteúdo volumétrico de água na zona de raiz profunda (28 a 100 cm).	Média das estimativas horárias.
Umidade do Solo (100–289 cm)	$m^3/m^3$	Conteúdo volumétrico de água na camada profunda (1 a 2,89 m).	Média das estimativas horárias.
Umidade relativa média (UR)	%	Obtida a partir de T2m e ponto de orvalho.	Derivada das médias horárias.
Vento a 10 m (V10m)	m/s	Módulo médio diário da velocidade do vento.	Média das componentes horárias (u,v).
Radiação solar incidente (Rs)	W/m <sup>2</sup>	Soma diária da radiação solar de onda curta que atinge a superfície.	Soma das estimativas horárias.
Temperatura mínima (Tmin)	°C	Menor valor diário da temperatura a 2 m.	Mínimo das estimativas horárias.
Temperatura máxima (Tmax)	°C	Maior valor diário da temperatura a 2 m.	Máximo das estimativas horárias.

Cada variável possui paleta e discretização próprias (ver Seção 5), garantindo coerência

visual entre produtos.

### C. Estrutura do repositório GitHub

O projeto oficial do CCC encontra-se em:

[https://github.com/Crepaldi2025/dashboard\\_cat314](https://github.com/Crepaldi2025/dashboard_cat314)

### D. Para enviar melhorias

1. Realize o `fork` do repositório;
2. Crie uma `branch` com o nome da modificação;
3. Descreva claramente a alteração e envie um `pull request`.

### E. Contatos e suporte

- **Autor:** Paulo César Crepaldi.
- **Orientador:** Prof. Dr. Enrique Vieira Mattos
- **E-mail de contato:** `crepaldi@unifei.edu.br`

## Bibliografia Recomendada

---

### Conjuntos de Dados (Reanálises)

---

MUÑOZ-SABATER, J. et al. ERA5-Land: A state-of-the-art global reanalysis dataset for land applications. *Earth System Science Data*, v. 13, n. 9, p. 4349–4383, 2021. DOI: <https://doi.org/10.5194/essd-13-4349-2021>.

HERSBACH, H. et al. The ERA5 global reanalysis. *Quarterly Journal of the Royal Meteorological Society*, v. 146, n. 730, p. 1999–2049, 2020. DOI: <https://doi.org/10.1002/qj.3803>.

OLAUSON, J. ERA5: The new champion of wind power modelling?. *Renewable Energy*, v. 126, p. 322–331, 2018. DOI: <https://doi.org/10.1016/j.renene.2018.03.056>.

COPERNICUS CLIMATE CHANGE SERVICE (C3S). *ERA5: Fifth generation of ECMWF atmospheric reanalyses of the global climate*. Copernicus Climate Change Service Climate Data Store (CDS), 2017. Disponível em: <https://cds.climate.copernicus.eu/>.

---

### Plataformas e Ferramentas

---

GORELICK, N. et al. Google Earth Engine: Planetary-scale geospatial analysis for everyone. *Remote Sensing of Environment*, v. 202, p. 18–27, 2017. DOI: <https://doi.org/10.1016/j.rse.2017.06.031>.

WU, Q. geemap: A Python package for interactive mapping with Google Earth Engine. *Journal of Open Source Software*, v. 5, n. 51, p. 2305, 2020. DOI: <https://doi.org/10.21105/joss.02305>.

PLOTLY TECHNOLOGIES INC. *Plotly: Collaborative data science*. Montréal, 2015. Disponível em: <https://plotly.com/>.

STREAMLIT INC. *Streamlit: The fastest way to build and share data apps*. 2024. Disponível em: <https://streamlit.io/>.

THE PANDAS DEVELOPMENT TEAM. pandas-dev/pandas: Pandas (Version 1.0.0). *Zenodo*, 2020. DOI: <https://doi.org/10.5281/zenodo.3509134>.

HUNTER, J. D. Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, v. 9, n. 3, p. 90–95, 2007. DOI: <https://doi.org/10.1109/MCSE.2007.55>.

---

### Computação em Nuvem e Documentação

---

WILSON, G. et al. Good enough practices in scientific computing. *PLoS Computational Biology*, v. 13, n. 6, e1005510, 2017. DOI: <https://doi.org/10.1371/journal.pcbi.1005510>.  
GOOGLE EARTH ENGINE TEAM. *Earth Engine Documentation*. Mountain View, CA, 2024.  
Disponível em: <https://developers.google.com/earth-engine>.

---

## Normas e Metodologias

---

ISO/IEC/IEEE. *ISO/IEC/IEEE 26514:2022 — Systems and software engineering — Design and development of information for users*. International Organization for Standardization, 2022.

WMO – World Meteorological Organization. *Guide to Meteorological Instruments and Methods of Observation*. 2023. Disponível em: <https://library.wmo.int/>.