

# Global Solution 2025.1 — Calor Extremo no Nordeste

Grupo: Erick Souza Pereira

Curso: Inteligência Artificial - FIAP

Data: 05/06/2025

Link do repositório GitHub: <https://github.com/leleozera/projetos-FIAPon/tree/Global-Solution-1-semester>

## Introdução

O aumento da intensidade e frequência dos eventos climáticos extremos, como ondas de calor, representa um desafio significativo para o Nordeste do Brasil. Estes fenômenos afetam a saúde da população, a produção agrícola e a disponibilidade de recursos hídricos. Este projeto propõe uma solução digital que simula a coleta, análise e previsão de risco de calor extremo, utilizando Machine Learning, banco de dados e lógica computacional, visando demonstrar como a tecnologia pode apoiar a prevenção e a tomada de decisão diante de desastres naturais.

## Desenvolvimento

### Descrição Geral da Solução

O sistema desenvolvido simula um cenário real onde sensores de temperatura e umidade captam dados ambientais. Estes dados são processados localmente por scripts Python, armazenados em um banco de dados SQLite e analisados por um modelo de Machine Learning que indica, em tempo real, o risco de calor extremo.

### Arquitetura da Solução

Fluxo:

1. Um script simula os dados do sensor (como se fosse um ESP32), gravando no banco de dados.
2. Um modelo de Machine Learning (Random Forest) é treinado a partir de dados históricos.
3. Outro script lê o dado mais recente do banco, faz a previsão e emite alerta.

Sugestão: Adicione um diagrama simples mostrando o fluxo dos arquivos, ou peça para eu criar um desenho para você!

## Tecnologias Utilizadas

- Python 3
- Pandas, scikit-learn, joblib (Machine Learning)
- SQLite (banco de dados local)
- Simulação do ESP32 via script Python
- Ambiente de desenvolvimento: PyCharm

## Descrição dos Componentes

1. Simulador do Sensor (simulador\_esp32.py): Gera valores aleatórios de temperatura e umidade, simulando a leitura de um sensor físico. Salva cada leitura no banco SQLite.
2. Banco de Dados (leitura.db): Estrutura simples, armazena: temperatura, umidade, data/hora.
3. Dados Históricos (historico\_calor.csv): Arquivo CSV com dados reais ou simulados do Nordeste, indicando casos de calor extremo (label 1) e situações normais (label 0).
4. Treinamento do Modelo (treinamento.py): Treina um modelo RandomForest para classificar risco de calor extremo com base nos dados históricos e salva o modelo (modelo\_calor\_extremo.pkl).
5. Análise e Alerta (analise.py): Lê a última leitura do banco, faz a previsão e exibe no terminal se há risco de calor extremo ou não.

## Integração das Disciplinas

- Machine Learning em Python: Treinamento, validação e uso do modelo de classificação.
- Banco de Dados: Utilização de SQLite para registro e consulta das leituras.
- Lógica de Programação: Scripts para simular hardware, processar dados, decidir e emitir alertas.
- Prototipação e simulação: Substituição do hardware real por scripts para garantir viabilidade mesmo sem ESP32.

## Resultados Esperados

O sistema identifica corretamente situações de calor extremo, emitindo alertas. Gera uma base de dados realista para simular decisões automatizadas. Permite fácil visualização e interpretação dos dados e dos alertas gerados.

Exemplo de saída no terminal:

ALERTA! Risco de calor extremo: 39.5°C / 22.0%

Temperatura normal: 31.0°C / 50.0%

!

## Conclusão

O projeto demonstra, de forma prática e didática, como a integração de conceitos de programação, banco de dados e inteligência artificial pode oferecer soluções reais para problemas de desastres naturais. Mesmo com a ausência do hardware, a simulação permite validar toda a lógica do sistema e comprovar a aplicabilidade da solução proposta.

## Referências

- [disasterscharter.org](https://disasterscharter.org) (base de inspiração dos dados históricos)
- Documentação oficial do scikit-learn, pandas, SQLite e Python