

Sistema Distribuído de Monitoramento de Condições Climáticas via APIs Públicas

Este sistema visa monitorar, em tempo real, as condições climáticas (temperatura, umidade, vento, chuva) de diferentes cidades usando uma API pública (OpenWeatherMap) (**Fonte de dados a decidir**). Coleta e envia periodicamente os dados para um servidor central, o qual armazena, processa e disponibiliza os dados em um painel web interativo. Além do monitoramento, o sistema é capaz de **realizar previsões climáticas**, utilizando os próprios dados coletados para treinar e alimentar um modelo de **Machine Learning** que:

- **Prevê a temperatura para a próxima hora.**
- **Classifica o tempo como “bom” ou “ruim”** com base em variáveis.
- **Estima a probabilidade de chuva** nas próximas horas.

(Parte de criação do modelo a definir conforme exploração inicial dos dados)

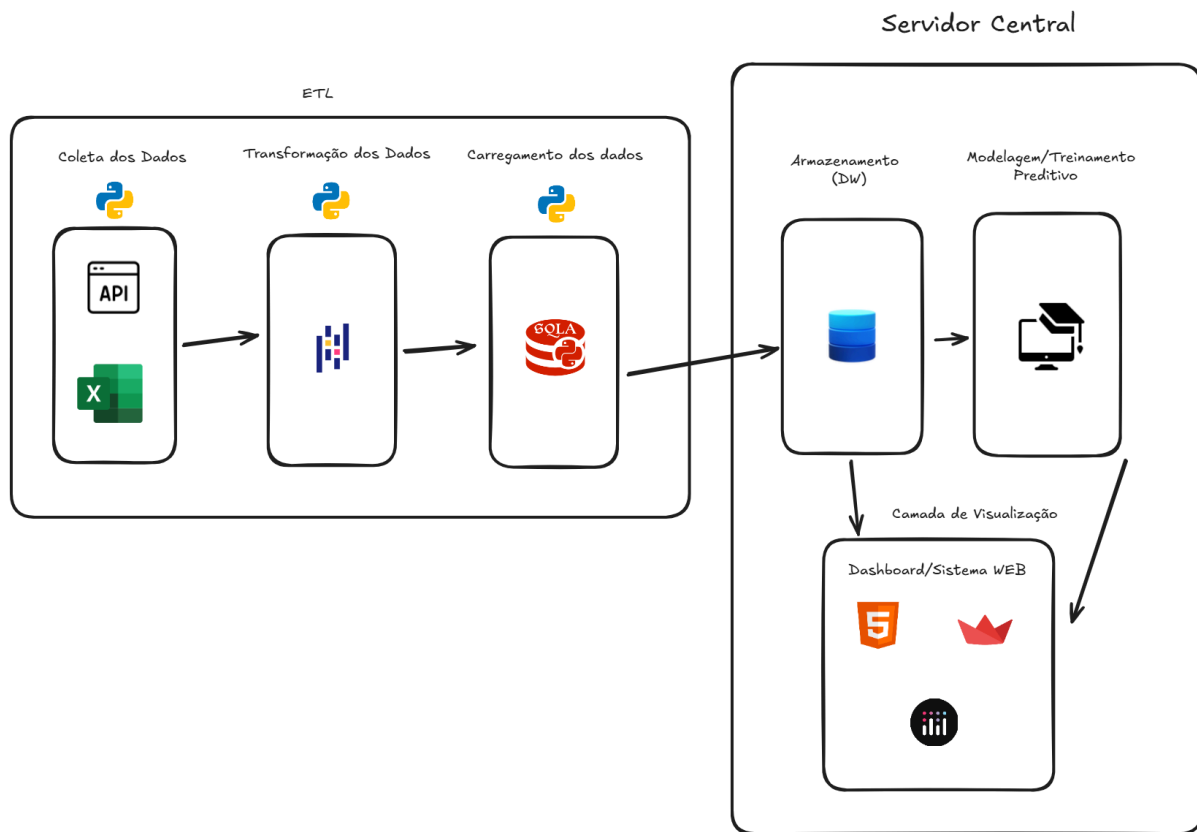
Entidades Principais:

- Script Coletor (ETL – Coleta dos Dados)
- Transformador de Dados (ETL – Transformação)
- Carregador de Dados (ETL – Carga)
- Data Warehouse (DW / Banco de Dados)
- Módulo de Modelagem Preditiva
- Modelo Treinado (.pkl)
- Servidor Central
- Dashboard Web (Camada de Visualização)

Metas de Sistemas Distribuídos Abordadas

- **Escalabilidade:** Um único coletor pode lidar com múltiplas cidades. É possível dividir os nós por grupo de cidades em servidores diferentes.
- **Disponibilidade:** Se um nó falhar, os demais continuam funcionando. O servidor central lida com perda de dados pontual.
- **Tolerância a Falhas:** O sistema pode armazenar dados localmente e reenviá-los. Logs de erro permitem tentativas.
- **Compartilhamento de Recursos:** Vários nós utilizam o mesmo servidor central e banco de dados. O modelo de ML é compartilhado para previsão em todas as cidades.

Arquitetura Inicial



Descrição Arquitetura

- **ETL (Extract, Transform, Load):**
Essa etapa é dividida em três componentes:
 - **Coleta dos Dados:** feita por meio de scripts em Python que realizam requisições a APIs públicas com dados climáticos.
 - **Transformação dos Dados:** responsável pelo tratamento, sanitização e preparação dos dados brutos.
 - **Carregamento dos Dados:** realiza a ingestão dos dados transformados em um formato previamente definido, com base em uma modelagem elaborada após a exploração inicial dos dados.
- **Armazenamento (DW):** Essa etapa é responsável por armazenar os dados em formato dimensional para facilitar a análise e dos dados, e terá uma versão de backup.

- **Modelagem/Treinamento Preditivo:** Essa etapa é responsável por toda parte de modelagem preditiva do modelo de ML aplicação técnicas necessárias para sua implementação.
- **Camada de Visualização dos dados:** Será a parte terá um Dashboard interativo no qual o usuário poderá ter acesso tanto aos dados de monitoramento histórico climáticos quanto às previsões climáticas.

Todos esses componentes terão comunicação com servidores.