

Desafio Técnico: Plataforma de Visualização de Dados do Agronegócio

Objetivo

Desenvolver uma aplicação que consuma dados públicos do agronegócio, processe-os via backend e os exiba em um dashboard interativo no frontend, com foco em visualização geográfica (mapa) e indicadores demográficos.

Entrega Esperada

- Backend com API REST que consome e trata dados públicos.
- Frontend com dashboard interativo (BI) com mapa e gráficos.
- Arquitetura em microsserviço.
- Vídeo: Demonstração da aplicação em funcionamento, explicação do código e raciocínio (upload no YouTube não listado, OneDrive ou Google Drive).
- Código-fonte no GitHub ou GitLab.

Requisitos Técnicos

Backend

- Linguagem: Python (FastAPI ou Flask), C# (.NET), Node.js (Javascript/Typescript).
- Funcionalidades:
 - Middleware que consome dados públicos da Produção Agrícola Municipal.
 - Processamento e normalização dos dados.
 - Exposição via API REST.
 - Microsserviço containerizado (Docker).

Frontend

- Linguagem: Javascript ou Typescript.
- Frameworks sugeridos: React, Vue ou Angular.
- Funcionalidades:
 - Dashboard com gráficos (ex: Chart.js, D3.js ou ECharts).
 - Mapa interativo (ex: Leaflet, Mapbox ou Google Maps).
 - Filtros por região, cultura agrícola, produção, etc.

Fontes de Dados (3 opções)

1. SIDRA - IBGE

- Consulta direta via URL e JSON.
- Exemplo:
<https://sidra.ibge.gov.br/pesquisa/pam/tabelas>
- Tabelas sugeridas:
 - 1612 – Lavouras temporárias
 - 1613 – Lavouras permanentes
 - 5457 – Consolidação

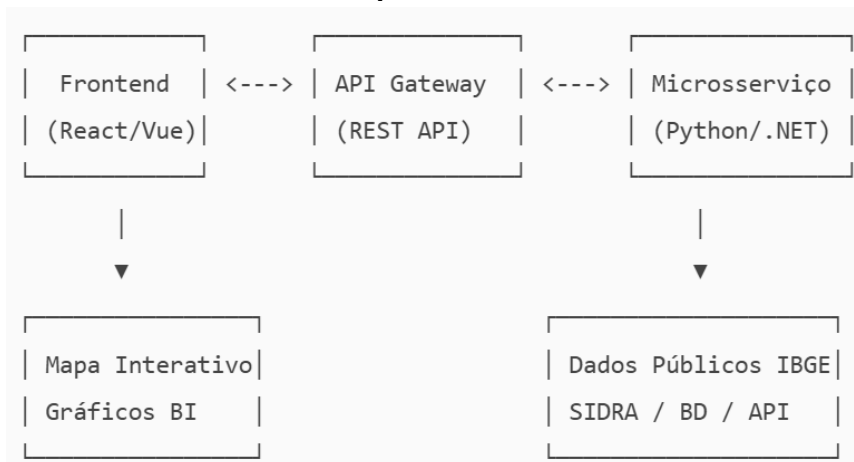
2. Base dos Dados + BigQuery

- Dados tratados e prontos para uso via SQL ou download.
- Link:
<https://basedosdados.org/dataset/fc403b40-a7e1-40e7-9efe-910847b45a69>

3. API do IBGE (Serviço de Dados)

- Documentação oficial:
<https://servicodados.ibge.gov.br/api/docs/>
- Pode ser usada para complementar dados geográficos e administrativos.

🧩 Desenho Básico da Arquitetura



📁 Formato da Entrega

- **Vídeo:** Demonstração da aplicação em funcionamento, explicação do código e raciocínio.
- **Código:** Disponibilizar via GitHub ou como anexo.
- **Link dos dados utilizados:** Informar no README.

✅ Critérios de Avaliação

- Clareza e organização do código.
- Estrutura da API e tratamento dos dados.
- Qualidade da visualização no frontend.
- Uso adequado de tecnologias.
- Explicação no vídeo.