

🎵 Nome e Descrição do Projeto

Nome do Projeto: Gestão de Instrumentos Musicais

Descrição:

A aplicação "Gestão de Instrumentos Musicais" é uma API RESTful desenvolvida em Django com Django REST Framework para gerenciar um catálogo de instrumentos musicais. Ela permite o cadastro, listagem e exclusão de instrumentos, mantendo os dados persistidos localmente em banco de dados SQLite. A interface front-end, feita com HTML, CSS e JavaScript puro, permite que o usuário interaja de forma visual com os dados.

🎯 Tema e Entidade Escolhida

Tema: Catálogo de instrumentos musicais

Entidade: Instrumento

📦 Estrutura dos Dados (Modelo JSON)

Cada instrumento é representado com os seguintes campos:

```
{
  "id": 1,
  "nome": "Violão",
  "categoria": "sopro",
  "preco": 2500,
  "descricao": "Instrumento de cordas acústico",
}
```

🌐 Lista de Rotas REST Utilizadas:

Método	Rota	Descrição
GET	/instruments/	Retorna todos os instrumentos
GET	/instruments/\:id/	Retorna um instrumento específico
POST	/instruments/	Adiciona um novo instrumento
DELETE	/instruments/\:id/	Remove um instrumento

🏗️ Estrutura Geral da Aplicação

Back-end (Django):

- Diretório: `/instruments`
 - `models.py`: define o modelo `Instrument`.
 - `serializers.py`: converte os modelos para JSON e vice-versa.
 - `views.py`: define as views com base nas `Generic Views`.
 - `urls.py`: roteamento das rotas da API.

Configuração principal:

- `music_api/settings.py`: configurações gerais do projeto.
- `db.sqlite3`: banco de dados local onde os dados são armazenados.
- `manage.py`: entrada principal da aplicação Django.

Front-end (HTML + JS puro):

- Diretório: `/frontend`
 - `index.html`: interface principal da aplicação.
 - `script.js`: lógica para consumir a API (com `fetch()`).
 - `style.css`: estilização básica da interface.

Integração com Front-End

O front-end é uma página simples em HTML que:

- Lista os instrumentos do banco de dados.
- Permite cadastrar novos instrumentos via formulário.
- Permite atualizar e excluir instrumentos utilizando JavaScript puro e `fetch()` para consumir a API.

 Código-Fonte da Aplicação:

Link do repositório: https://github.com/Rafael1711/Atv-1_BACK