

Projeto de Persistência Poliglota com MongoDB + SQLite e Geo-Processamento em Python com Streamlit

Objetivo:

Desenvolver uma aplicação prática utilizando **Persistência Poliglota** (SQLite + MongoDB) para armazenar e consultar dados em diferentes contextos. Além disso, a aplicação deve implementar **recursos de geoprocessamento** utilizando objetos do tipo **JSON** para manipular e visualizar dados espaciais (latitude e longitude).

Requisitos do Projeto:

1. Banco de Dados

- **SQLite:**
 - Armazenar dados tabulares estruturados (exemplo: informações de usuários, cidades, estados, países).
- **MongoDB:**
 - Armazenar documentos no formato **JSON**, incluindo **coordenadas geográficas** (latitude, longitude) e metadados associados.
 - Exemplo de documento:

```
{
  "nome_local": "Praça da Independência",
  "cidade": "João Pessoa",
  "coordenadas": {
    "latitude": -7.11532,
    "longitude": -34.861
  },
  "descricao": "Ponto turístico central da cidade."
}
```

2. GEOProcessamento

- Implementar funções em **Python** para:
 - Calcular distâncias entre dois pontos geográficos.
 - Listar os locais em um **raio de distância** a partir de uma coordenada fornecida.
 - Consultar dados armazenados no MongoDB em formato JSON e cruzar informações com dados no SQLite.

3. Interface (Streamlit)

- A aplicação deve ter uma interface **interativa** que permita:
 - Inserir novos dados no **SQLite** e no **MongoDB**.
 - Selecionar uma cidade/estado do **SQLite** e visualizar os pontos de interesse armazenados no **MongoDB**.
 - Mostrar no mapa (via `st.map()` ou bibliotecas como **folium/pydeck**) os locais armazenados.

- Exibir resultados das consultas de geoprocessamento (por exemplo: locais em até 10 km de uma coordenada).

4. Funcionalidades Esperadas

1. **Cadastro de locais** com informações básicas e coordenadas (MongoDB).
2. **Cadastro de cidades/estados** em tabela (SQLite).
3. **Consulta integrada:** exibir locais do MongoDB relacionados a uma cidade do SQLite.
4. **Função de proximidade geográfica:** dado um ponto (latitude/longitude), listar os locais próximos.
5. **Visualização no mapa** dentro do Streamlit.

Estrutura do Projeto (Sugestão)

```
projeto_persistencia_poliglota/  
├── app.py                # Aplicação principal em Streamlit  
├── db_sqlite.py          # Conexão e funções do SQLite  
├── db_mongo.py           # Conexão e funções do MongoDB  
├── geoprocessamento.py  # Funções de cálculo geográfico  
└── requirements.txt      # Bibliotecas necessárias
```

Tecnologias e Bibliotecas

- **Python 3.10+**
- **Streamlit** (interface)
- **SQLite3** (banco relacional)
- **PyMongo** (integração com MongoDB)
- **Geopy** (cálculo de distâncias)
- **Folium ou Pydeck** (visualização no mapa)
- **Pandas** (manipulação de dados)

Entregáveis

- Código fonte completo do projeto.
 - Documento (Markdown ou PDF) com:
 - Explicação da arquitetura adotada.
 - Exemplos de consultas realizadas.
 - Prints da interface em execução.
-

Critérios de Avaliação

- (+2,0 pts extras) Projeto entregue de forma completa, funcional e bem documentada.

Esse projeto ajudará os alunos a praticarem:

- Integração de diferentes tipos de bancos de dados (**persistência poliglota**).
- Manipulação de objetos JSON no MongoDB.
- Geoprocessamento e cálculo de distâncias.
- Construção de dashboards interativos no **Streamlit**.

Professor: Ricardo Roberto de Lima.

Data de Entrega: Dia da Prova AV1 até 23h59min.