

Relatório do Projeto: Consultor de Banco de Dados com IA

Autor: Bruno Gutierrez Mattarazzo de Souza

1. Descrição Geral do Projeto

Este projeto é uma aplicação web interativa, construída com a biblioteca Streamlit, que atua como uma interface inteligente para bancos de dados. O sistema permite que um usuário faça perguntas em linguagem natural (português) sobre os dados contidos em um banco de dados MySQL ou PostgreSQL. A aplicação utiliza um modelo de linguagem avançado (Llama 3, via API da Groq) para traduzir a pergunta em uma consulta SQL válida, executá-la no banco de dados selecionado e, por fim, apresentar o resultado de forma clara e formatada para o usuário.

2. Funcionamento da Aplicação

O fluxo de trabalho da aplicação ocorre em algumas etapas principais:

1. **Conexão com o Banco de Dados:** O usuário escolhe entre MySQL ou PostgreSQL. A aplicação utiliza as credenciais armazenadas no arquivo de ambiente (`.env`) para estabelecer uma conexão segura.
2. **Entrada do Usuário:** O usuário digita uma pergunta em uma caixa de texto (ex: "Quais são os 5 produtos mais caros?").
3. **Tradução para SQL:** A pergunta do usuário, juntamente com o esquema (a estrutura das tabelas) do banco de dados, é enviada para a primeira cadeia de IA. Esta IA, especializada em SQL, gera uma consulta SQL precisa para responder à pergunta.
4. **Execução da Consulta:** A consulta SQL gerada é executada diretamente no banco de dados conectado, retornando os dados brutos.
5. **Formatação da Resposta:** Os dados brutos são enviados para uma segunda cadeia de IA, que tem a única tarefa de formatar esses dados em um texto em português, claro e de fácil leitura, respondendo diretamente à pergunta original do usuário.
6. **Exibição dos Resultados:** A interface exibe a resposta formatada, uma tabela com os dados brutos e a consulta SQL que foi gerada, permitindo que o usuário valide todas as etapas do processo.

3. Configuração do Banco de Dados (`.env`)

Para que a aplicação se conecte a um banco de dados específico, todas as configurações de conexão são gerenciadas em um arquivo de ambiente chamado `.env`. Este arquivo deve ser criado na pasta principal do projeto e serve para armazenar informações sensíveis, como senhas e chaves de API, de forma segura e separada do código-fonte.

O arquivo deve conter as seguintes variáveis:

- **GROQ_API_KEY:** Chave de acesso para a API da Groq, necessária para o funcionamento da inteligência artificial.
- **MYSQL_URI** ou **POSTGRES_URI:** String de conexão que contém todas as informações para acessar o banco de dados.

Estrutura do arquivo `.env`:

Snippet de código

Python

```
# Adicione aqui sua chave de API gratuita do Groq

GROQ_API_KEY="SUA_CHAVE_GROQ_API_AQUI"


# Configure a string de conexão para seu banco MySQL

MYSQL_URI="mysql+mysqlconnector://SEU_USUARIO:SUA_SENHA@ENDereco_DO_SERVIDOR:PORTA/NOME_DO_BANCO"


# Configure a string de conexão para seu banco PostgreSQL

POSTGRES_URI="postgresql://SEU_USUARIO:SUA_SENHA@ENDereco_DO_SERVIDOR:PORTA/NOME_DO_BANCO"
```

Como preencher a URI do banco de dados:

- **SEU_USUARIO**: Substitua pelo nome de usuário do seu banco de dados (ex: **root** para MySQL).
- **SUA_SENHA**: Substitua pela senha de acesso ao banco.
- **ENDereco_DO_SERVIDOR**: Se o banco estiver rodando na mesma máquina que a aplicação, use **localhost**. Caso contrário, use o endereço IP do servidor.
- **PORTA**: A porta padrão é **3306** para MySQL e **5432** para PostgreSQL.
- **NOME_DO_BANCO**: O nome exato do banco de dados que será consultado.

4. Como Executar o Projeto (Guia de Instalação)

Pré-requisitos

- Python 3.8 ou superior.
- Acesso a um banco de dados MySQL ou PostgreSQL.

Passo 1: Preparar a Pasta do Projeto

Descompacte os arquivos do projeto em uma pasta. Abra seu terminal (ou Prompt de Comando) e navegue até o diretório do projeto usando o comando **cd**.

Bash

Python

```
# Exemplo de como navegar para a pasta
```

```
cd caminho/para/a/pasta_do_projeto
```

Passo 2: Criar e Ativar o Ambiente Virtual

É uma boa prática isolar as dependências do projeto.

Bash

Python

Comando para criar o ambiente

```
python3 -m venv .venv
```

- No **macOS / Linux**: `source .venv/bin/activate`
- No **Windows**: `.venv\Scripts\activate`

Passo 3: Instalar as Bibliotecas

Com o ambiente virtual ativo, instale todas as dependências necessárias.

Bash

Python

```
pip install streamlit pandas python-dotenv langchain-community  
langchain-groq psycopg2-binary mysql-connector-python
```

Passo 4: Configurar o Banco de Dados

Crie e edite o arquivo `.env` na pasta do projeto, conforme detalhado na **Seção 3** deste relatório.

Passo 5: Rodar a Aplicação

Finalmente, execute o seguinte comando no terminal:

Bash

Python

```
streamlit run app.py
```

A aplicação será iniciada em uma nova aba do seu navegador.