Atelier: coder avec une IA générative

Utilisation de Large Language Models pour développer un logiciel en respectant les standards de l'industrie

Partie 4

Nicolas Debeissat
nicolas.debeissat@mail-formateur.net

Plan du module

Installation de l’extension continue.dev

Dans VSCode, avec ollama en local

Utiliser la configuration de base

Création d’une stack React + FastAPI + Postgresql

Configuration avancée et codebase

Comment adapter ses réponses

RAG avec Langchain

API de génération de requêtes ElasticSearch

Model Context Protocol (mode Agent)

Création d’outils et connexion à son agent

Evaluation

Quizz et rendu du projet effectué en cours

RAG avec Langchain - créer le projet

- https://templates.langchain.com/?integration_name=elastic-query-generator
- Créer un répertoire pour le projet

- poetry init
- poetry config virtualenvs.in-project true --local
- modifier : requires-python = ">=3.12,<4.0" dans pyproject.toml
- poetry add langchain-cli
- source .venv/Scripts/activate
- langchain app new my-app
- cd ./my-app
- langchain app add elastic-query-generator --branch v0.2

RAG avec Langchain - servir le projet

- code .
- deactivate
- poetry config virtualenvs.in-project true --local
- poetry install
- source .venv/Scripts/activate
- poetry add httpx=0.27.2
- dans server.py, à la place de add_routes(app, NotImplemented) :

```
from elastic_query_generator.chain import chain as elastic_query_generator_chain
add_routes(app, elastic_query_generator_chain, path="/elastic-query-generator")
```

- langchain serve

RAG avec Langchain - connections

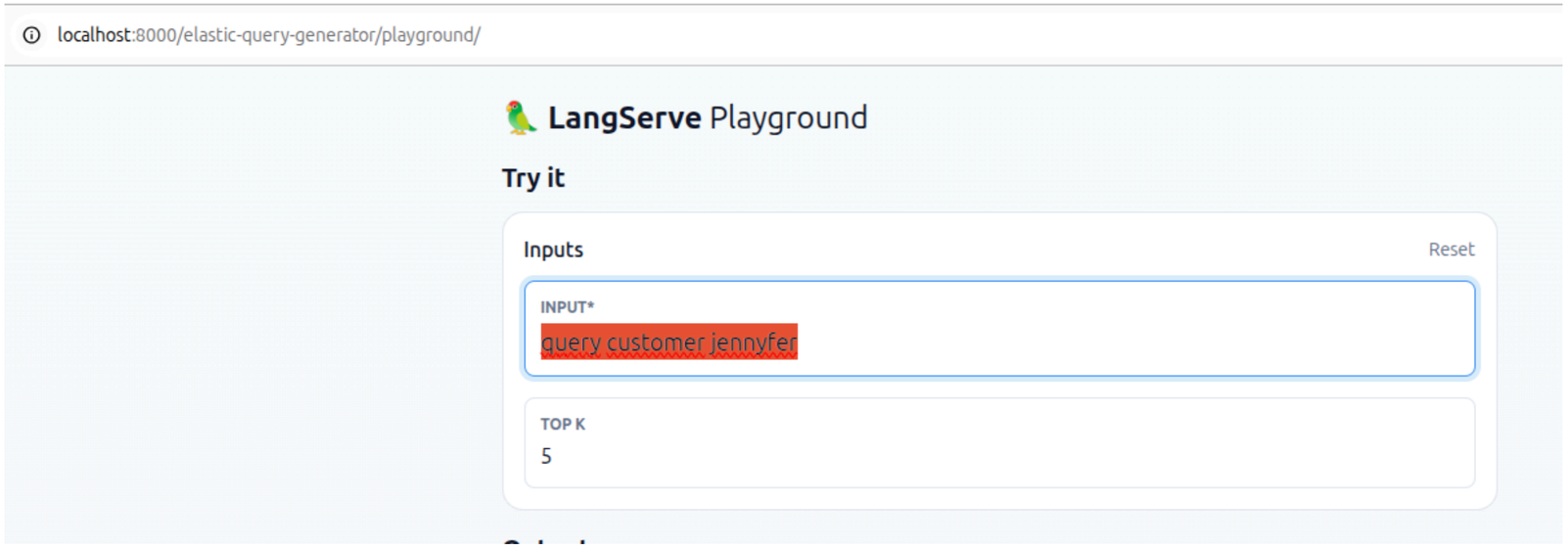
ELASTIC_PASSWORD = "..."
CLOUD_ID = "..."
db = Elasticsearch(cloud_id=CLOUD_ID, basic_auth=("elastic",
ELASTIC_PASSWORD))
→
db = Elasticsearch(cloud_id=CLOUD_ID, api_key="")

et
_model = ChatOpenAI(temperature=0, model="gpt-4")
->
_model = ChatOllama(model="deepseek-r1:1.5b")

RAG avec Langchain - ingest.py

python packages/elastic-query-generator/ingest.py

http://localhost:8000/elastic-query-generator/playground/



RAG avec Langchain

Beaucoup d’autres exemples à parcourir :

https://github.com/langchain-ai/langchain/tree/v0.2/templates/