

Progetto finale - Laboratorio di Ingegneria Informatica

Text-to-SQL con Ollama e FastAPI

Thomas Casu 1960733– Sapienza Università di Roma

Obiettivo del progetto

Il progetto consiste nella realizzazione di una API (Application Programming Interface) che permette all'utente di interrogare un database movies tramite il semplice utilizzo del linguaggio naturale. Ad esempio il server, ricevuta una domanda, la converte in una query SQL usando un LLM (Large Language Model) eseguito localmente con Ollama, restituendo i risultati dell'esecuzione della query su un database MariaDB.

Il sistema è progettato per essere modulare, scalabile e conforme agli standard REST, integrando anche un'interfaccia HTML per l'interazione.

Schema del database

Il database relazionale contiene una singola tabella, coerentemente con i dati assegnati, chiamata *movies*. Le colonne della tabella, e i rispettivi domini, sono le seguenti:

- id: intero, chiave primaria, not null
 - titolo: stringa, not null
 - regista: stringa, not null
 - eta_autore: intero, not null
 - anno: intero, not null
 - genere: stringa, not null
 - piattaforma_1: stringa, not null
 - piattaforma_2: stringa
-

Organizzazione del codice

Il progetto è suddiviso in moduli, ciascuno con responsabilità specifiche:

- **backend/src/api**: definisce gli endpoint REST:
 - `/router_add` per aggiungere un nuovo film;
 - `/router_schema` per restituire lo schema del database;
 - `/router_search` per convertire la domanda in SQL ed eseguire la query;
 - `/router_sql` per eseguire query SQL fornite direttamente.
 - **backend/src/core**:
 - `db.py`: gestione della connessione a MariaDB, importato nella maggior parte degli endpoint;
 - `ollama_client.py`: invio richieste a Ollama per la generazione SQL;
 - **backend/src/models**: modelli Pydantic per la validazione dei dati in ingresso e uscita;
 - **frontend**: interfaccia HTML basata su template Jinja2;
 - **mariadb**: script SQL e dati iniziali;
 - **docker-compose.yml**: file per l'orchestrazione dei container (MariaDB, backend, frontend, Ollama).
-

Funzionamento degli endpoint

- **GET /schema_summary**
Restituisce la struttura del database in JSON (elenco delle tabelle e le rispettive colonne).
- **POST /add**
Aggiorna il database con l'aggiunta di un nuovo film partendo da una stringa in input. Il film aggiunto viene memorizzato e sarà presente anche in caso di riavvio del server.
- **POST /search**

Invia una domanda all'LLM (Ollama), ottiene una query SQL, la esegue se è un SELECT valido e restituisce i risultati.

- **POST /sql_search**

Esegue direttamente una query SQL fornita dall'utente. Se la query non è un SELECT, viene rifiutata come unsafe.

Note finali

Il progetto è stato verificato tramite lo script di test assegnato e test manuali in corso di sviluppo per assicurare il corretto funzionamento degli endpoint ad ogni fase, e la conformità al formato di input/output richiesto. L'architettura modulare è stata applicata con il fine di facilitare la leggibilità ed eventuali estensioni future.