RAG-SupportBot: Mimari Dokümantasyon

Bu belge, cihaz üstünde (offline) veya online çalışan, belge tabanlı teknik destek asistanı olan **RAG-SupportBot** projesinin mimari kararlarını, kullanılan teknolojileri ve bileşenlerin işleyişini detaylı şekilde açıklamak amacıyla hazırlanmıştır.

Kullanılan Ana Modeller

Embedding Modeli: `sentence-transformers/all-MiniLM-L6-v2`

Bu model, küçük boyutlu, hızlı ve semantic search kalitesi açısından dengeli sonuçlar verir.
MVP aşamasında **yüksek hız** ve **düşük kaynak tüketimi** sağladığı için tercih edilmiştir.
Daha sonra `nomic-embed-text-v1` veya `instructor-xl` gibi daha güçlü embedding modellerine geçilmesi planlanmaktadır.

LLM Modeli: `meta-llama/llama-3-8b-instruct`

- **Çalışma şekli:** OpenRouter API üzerinden online olarak çalıştırılır.
- **Neden seçildi:** RAM sınırlamaları nedeniyle LLM yerelde çalıştırılmamış, API kullanımı ile donanım kısıtı aşılmıştır.
- **Avantaj:** Farklı LLM modellerine hızlı geçiş imkanı.
- **Geliştirme önerisi:** Model adı `.env` üzerinden yönetilebilir hale getirilmeli.

Vector Store: FAISS + LlamaIndex

FAISS, yüksek performanslı bir vektör benzerlik arama kütüphanesidir.
LlamaIndex ile entegre edilerek belge parçalarının vektörleştirilmiş halleri saklanır ve sorgulara göre en benzer içerikler bulunur.
Veriler `db/faiss_index` klasöründe JSON tabanlı formatta saklanır.

MVP Modülleri

`embedder.py`

- `data/` klasöründeki belgeleri okur.
- Belgeleri **500 token**'lık anlamlı parçalara ayırır.
- Her parçanın embedding vektörünü üretir.
- FAISS vektör veritabanına kaydeder.
- Data değiştiğinde tekrar çalıştırılarak FAISS güncellenir.

`retriever.py`

- Kullanıcıdan gelen sorunun embedding vektörünü üretir.
- FAISS veritabanından **Top-K** en benzer içerikleri getirir.
- **Reranker** ile bu parçalar, sorguya göre yeniden sıralanır.
- Yalnızca en alakalı içerikler LLM'e iletilir.
- Yanlış cevap verme riski azalır, doğruluk artar.

`llm_generator.py`

- Retriever'dan gelen belgeler ve kullanıcı sorusunu alır.
- **OpenRouter API** aracılığıyla LLM'e gönderir.
- Prompt yalnızca verilen belgelerden cevap üretmeyi garanti edecek şekilde tasarlanır.
- Üretilen cevap `last_answer.txt` dosyasına kaydedilir.
- **İyileştirme önerisi:** Prompt versiyonlama eklenerek değişiklikler takip edilebilir.

Gelecek Genişletmeler

- `query_rewriter.py`: Eksik veya kısa soruları iyileştirerek daha iyi arama sonuçları üretir.
- `document_grader.py`: Getirilen belgelerin kalite kontrolünü yapar.
- `fallback_router.py`: Cevap bulunamazsa alternatif işlem akışlarını yönetir.
- Prompt versiyonlama: Prompt değişikliklerinin ayrı dosyada tutulması.
