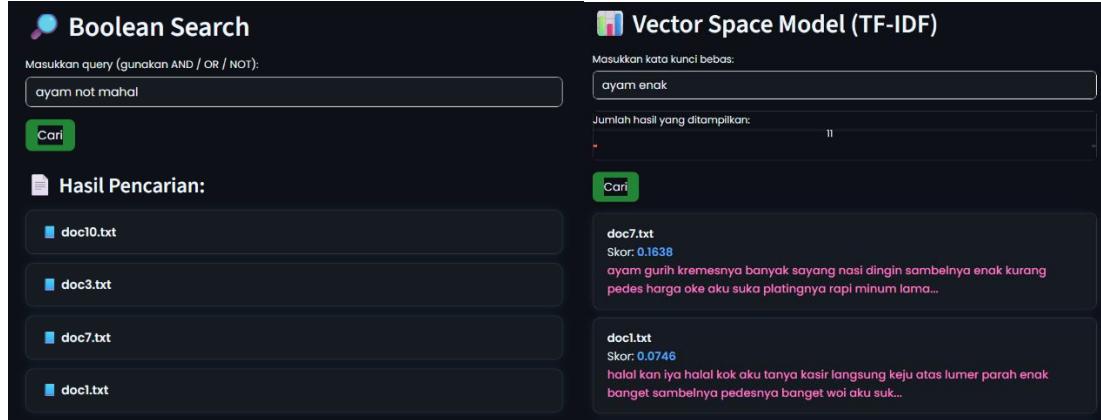


ESAI DAN DEMO ARSITEKTUR IR

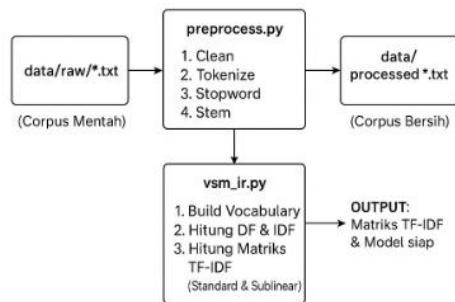
Boolean model dan Vsm model



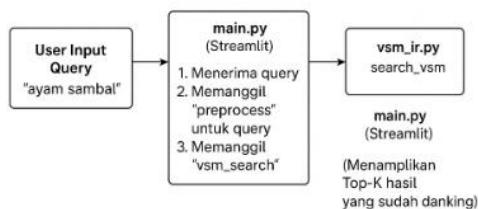
1. **Sistem Temu Kembali Informasi** (STKI) dirancang untuk bekerja dengan data tidak terstruktur (seperti dokumen teks, halaman web, atau ulasan) dan melayani kebutuhan informasi (information need) yang seringkali bersifat *ambigu*.
2. **Garis besar arsitektur**
 - o **Fase Offline (Indexing)**, yang berjalan satu kali untuk mempersiapkan data. Fase ini mencakup:
Data Collection: Mengumpulkan dokumen (di Proyek M-SE, ini adalah file data/raw/).
Document Preprocessing: Membersihkan data mentah (via preprocess.py: *cleaning, tokenizing, stopword removal, stemming*) menjadi corpus bersih (data/processed/).
Indexing: Membangun struktur data pencarian cepat (via vsm_ir.py: *Vocabulary, IDF*, dan Matriks TF-IDF).
 - o **Fase Online (Querying)**, yang berjalan *real-time* saat pengguna mencari:
Query Input: Pengguna memasukkan *query* (via main.py Streamlit).
Query Preprocessing: *Query* dibersihkan menggunakan alur yang sama dengan dokumen.
Retrieval & Ranking: Sistem menghitung vektor TF-IDF untuk *query* dan membandingkannya dengan semua vektor dokumen (menggunakan *Cosine Similarity*) untuk mendapatkan skor.
Presentation: Hasil Top-K yang sudah di-ranking disajikan kepada pengguna.

3. Sketsa Arsitektur

FASE OFFLINE (Indexing) -
Dijalankan satu kali saat aplikasi di-load



FASE ONLINE (Querying) -
Dijalankan setiap kali user menekan "Cari"



4. Relevansi Proyek

Soal 02 (Preprocessing / Materi 2) -> Sub-CPMK 10.1.2: Ini diimplementasikan secara penuh dalam modul preprocess.py. Modul ini bertanggung jawab atas seluruh tahapan *document preprocessing*, mulai dari membaca data mentah, melakukan *cleaning* (case folding, punctuation removal), *tokenizing*, *stopword removal* (Sastrawi), hingga *stemming* (Sastrawi).

Soal 03 (Pemodelan / Materi 3) -> Sub-CPMK 10.1.3: Ini diimplementasikan dalam dua modul inti: boolean_ir.py (tidak diserahkan, namun diimpor) yang menangani *Boolean Retrieval Model*, dan vsm_ir.py yang secara spesifik mengimplementasikan *Vector Space Model*.

Soal 04 (Term Weighting / Materi 4) -> Sub-CPMK 10.1.4 (Bagian 1): Ini adalah inti dari vsm_ir.py. Proyek ini tidak hanya menerapkan VSM, tetapi secara spesifik mengimplementasikan konsep *Term Weighting*. Ini dibuktikan dengan adanya dua skema berbeda: compute_tfidf_standard (raw TF) dan compute_tfidf_sublinear (1 + log TF), yang menunjukkan pemahaman mendalam tentang bagaimana *weighting* memengaruhi model.

Soal 05 (Evaluasi / Materi 5-7) -> Sub-CPMK 10.1.4 (Bagian 2): Ini diimplementasikan dalam modul eval.py dan vsm_ir.py. Proyek ini tidak berhenti pada implementasi, tetapi juga melakukan *evaluasi model* secara kuantitatif.