

Artificial Intelligence

Laporan Progres Proyek Minggu ke-3



Disusun Oleh:

Andreas Teguh Santoso Kosasih - 140810230047

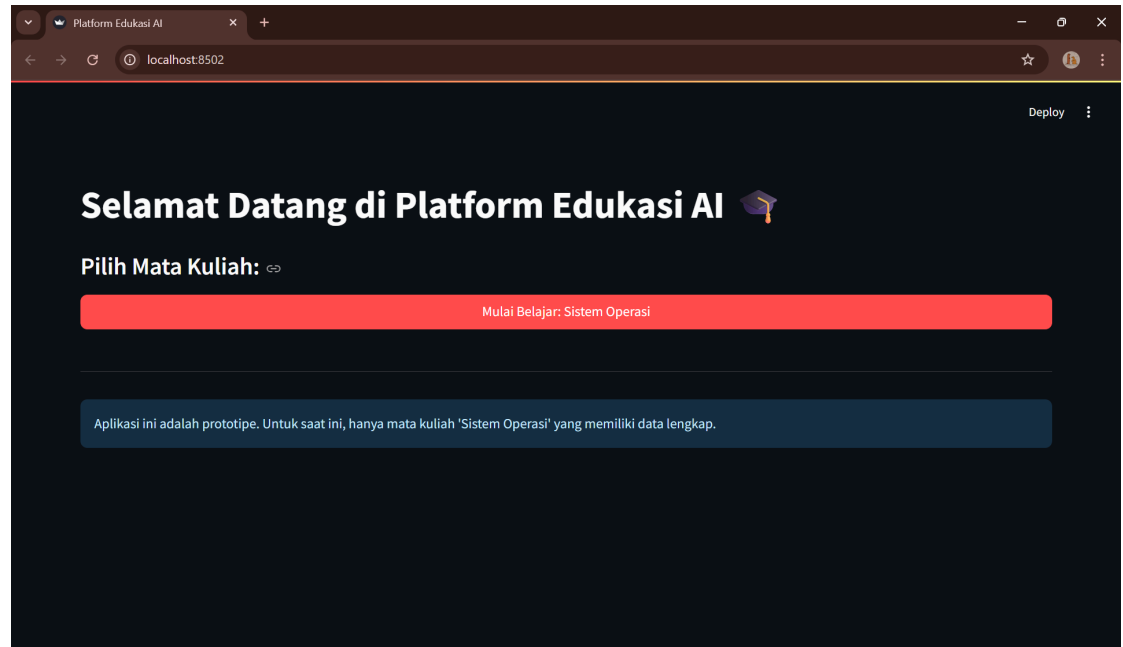
Muhammad Raihan Rizky Zain - 140810230049

Atharik Putra Rajendra - 140810230077

**PROGRAM STUDI S1 TEKNIK INFORMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS PADJADJARAN
JATINANGOR
2025**

1. Demo Aplikasi

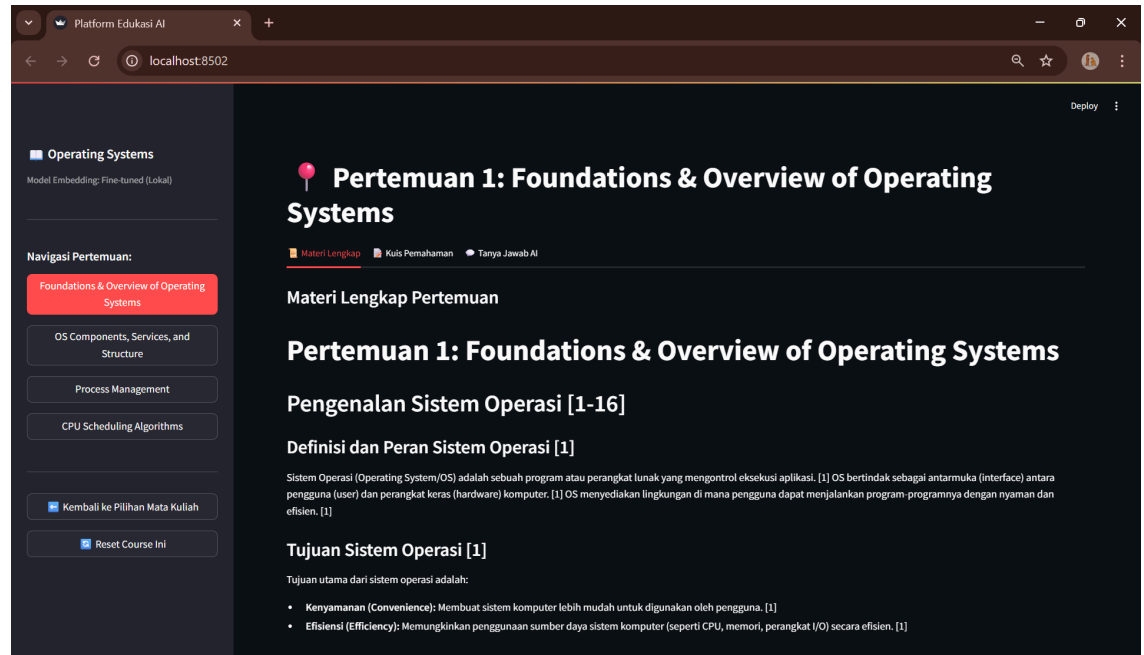
a. Tampilan Awal



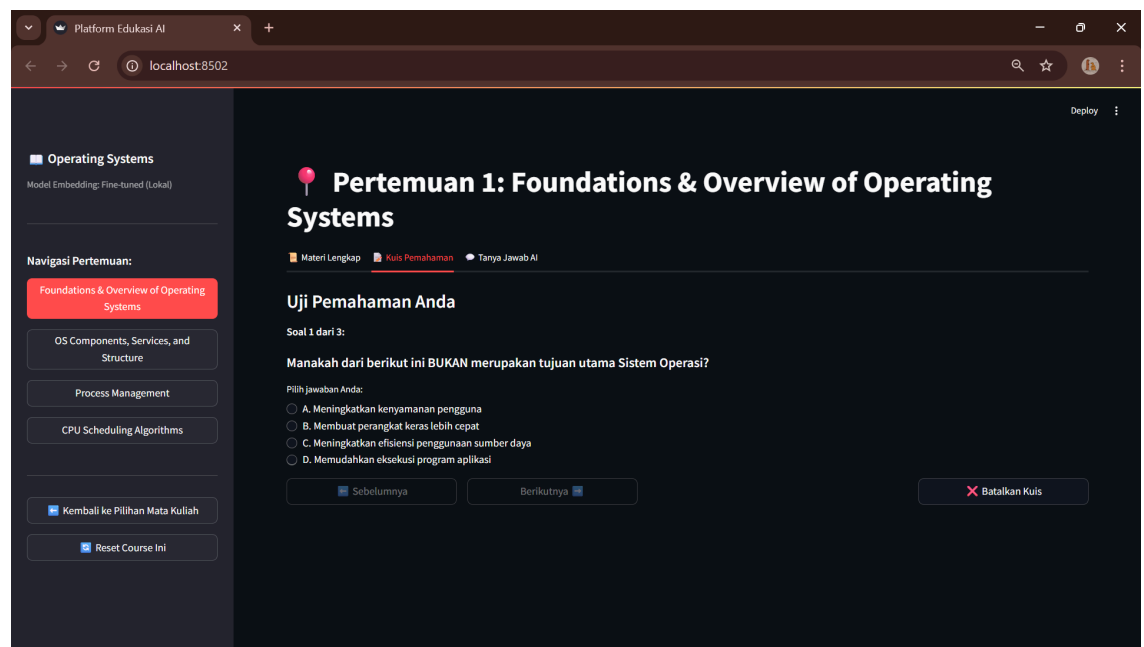
b. Main Menu

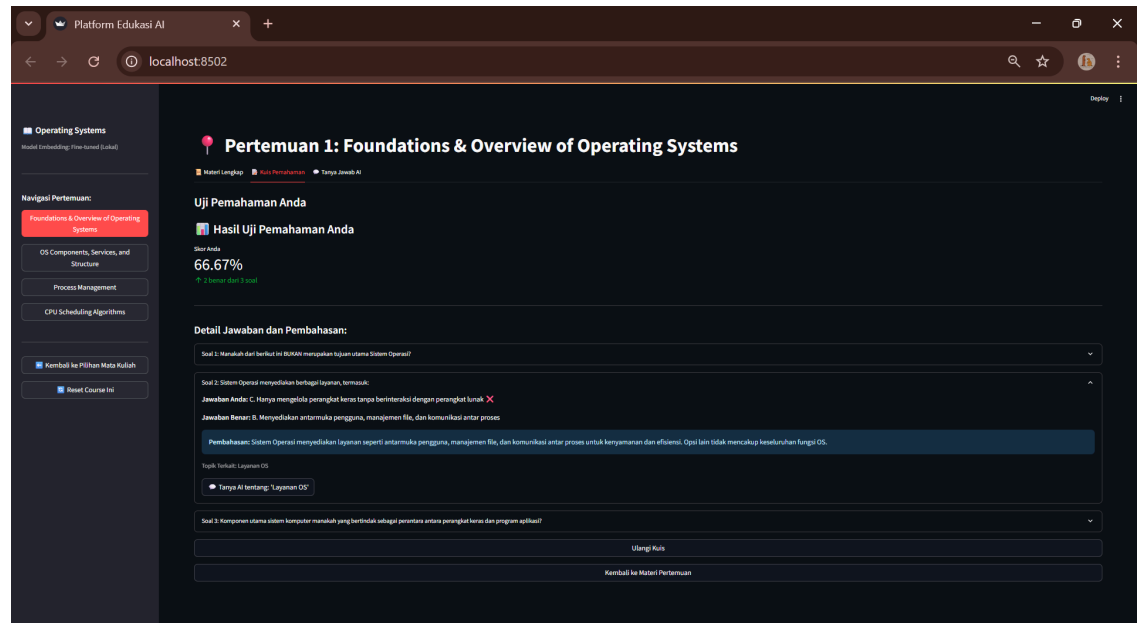


c. Menu Course Matkul (Materi Lengkap)

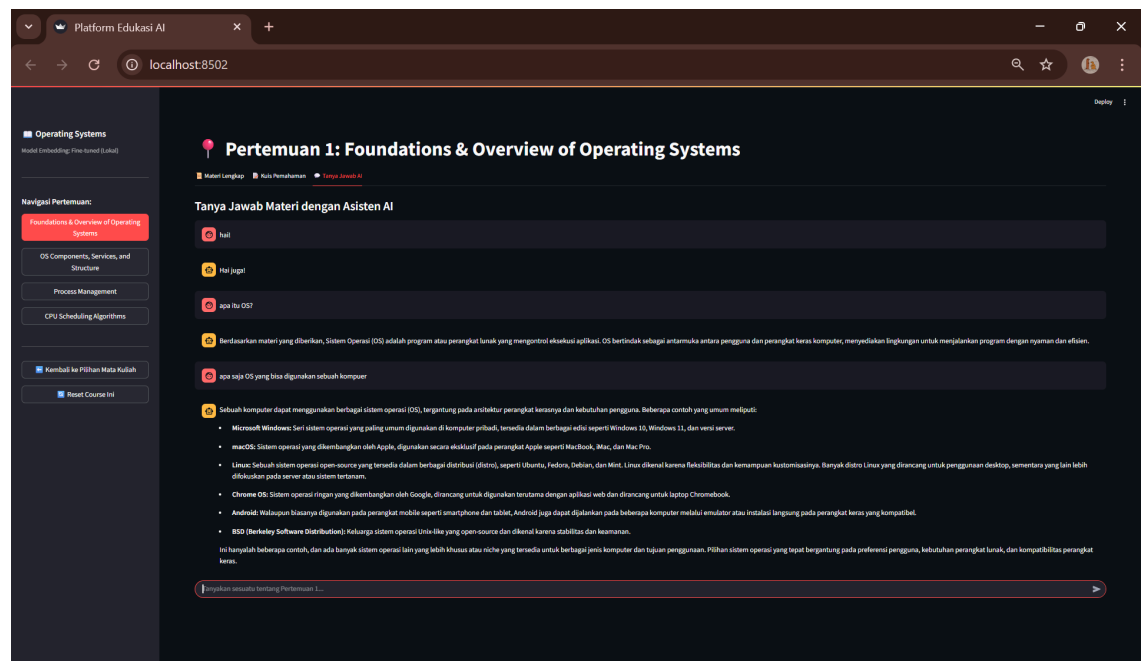


d. Menu Course Matkul (Chatbot Quiz)





e. Menu Course Matkul (Chatbot Tanya Jawab)



2. Visualisasi Model

Pipeline (Alur Kerja) Model

Pipeline model ini terdiri dari beberapa tahap utama:

1. Persiapan Data:

- Materi kuliah dari berbagai pertemuan diurai dan dipecah menjadi potongan-potongan teks yang lebih kecil (chunks). Proses ini penting agar informasi bisa diindeks dan diambil secara efisien. Total 56 chunk dihasilkan dari 4 pertemuan.
- Setiap chunk teks kemudian diubah menjadi representasi numerik (embedding) menggunakan model embedding dasar, yaitu `all-MiniLM-L6-v2`.
- Embedding ini disimpan dalam sebuah indeks FAISS (`vector_store_base.index`) untuk pencarian cepat, dan metadata chunk disimpan dalam file JSON (`processed_chunks_metadata_base.json`).

2. Fine-Tuning Model Embedding (Opsional):

- Jika tersedia dataset fine-tuning (`embedding_finetuning_examples.jsonl`), model embedding dasar (`all-MiniLM-L6-v2`) akan dilatih lebih lanjut (fine-tuned) menggunakan dataset tersebut. Tujuannya adalah agar model lebih baik dalam memahami konteks spesifik materi kuliah.
- Model yang sudah di-fine-tune kemudian disimpan (`finetuned_embedding_model_sistem_operasi`) dan digunakan untuk membuat ulang indeks FAISS (`vector_store_finetuned.index`) serta metadata chunk (`processed_chunks_metadata_finetuned.json`).

3. Retrieval Augmented Generation (RAG) untuk Chatbot:

- Saat pengguna bertanya di aplikasi Streamlit, pertanyaan tersebut diubah menjadi embedding menggunakan model embedding yang aktif (fine-tuned jika ada, jika tidak, model dasar).
- Embedding pertanyaan ini digunakan untuk mencari chunk materi yang paling relevan dari indeks FAISS.
- Chunk-chunk relevan ini kemudian diberikan sebagai konteks tambahan kepada Large Language Model (LLM) Gemini (**gemini-1.5-flash-latest**) bersama dengan pertanyaan asli pengguna.
- LLM Gemini menghasilkan jawaban berdasarkan konteks yang diberikan.

4. Generasi Kuis:

- LLM Gemini juga digunakan untuk membuat soal pilihan ganda berdasarkan materi pertemuan yang dipilih pengguna. Konteks dari chunk materi pertemuan tersebut diberikan kepada LLM untuk menghasilkan pertanyaan, opsi jawaban, jawaban benar, pembahasan, dan topik terkait.

Konsep "Learning" (Pembelajaran)

Konsep pembelajaran dalam sistem Anda terjadi terutama melalui fine-tuning model embedding:

1. Transfer Learning Awal: Model dasar **all-MiniLM-L6-v2** adalah

model yang sudah pra-terlatih pada dataset besar, sehingga ia sudah memiliki pemahaman umum tentang bahasa. Penggunaan model ini adalah bentuk transfer learning.

2. Fine-Tuning (Pembelajaran Spesifik Domain):

- Dengan menyediakan pasangan contoh (`query`, `positive_passage`) dari materi "Sistem Operasi" Anda dalam file `embedding_finetuning_examples.jsonl`, model embedding diajarkan untuk lebih memahami hubungan semantik *khusus* dalam domain tersebut.
- Model belajar untuk menghasilkan embedding yang lebih mirip untuk query dan passage yang memang relevan dalam konteks materi Anda. Ini dilakukan dengan menggunakan loss function seperti `MultipleNegativesRankingLoss`, yang bertujuan untuk membuat embedding dari pasangan positif lebih dekat satu sama lain di ruang vektor dibandingkan dengan pasangan negatif (yang secara implisit diambil dari sampel lain dalam batch).
- Proses ini mengadaptasi "pengetahuan umum" model ke "pengetahuan spesifik" mata kuliah Sistem Operasi.

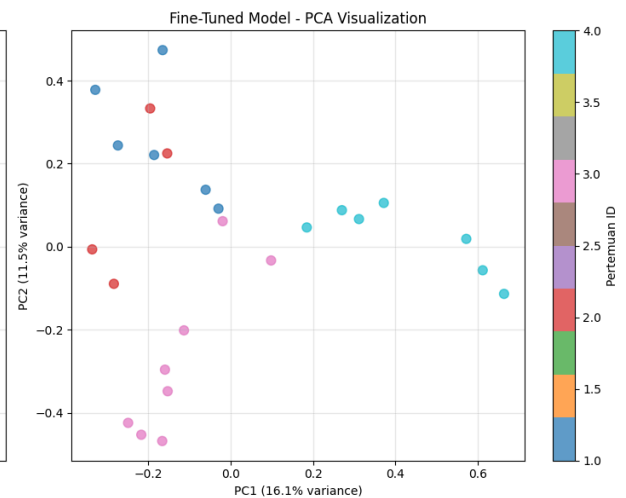
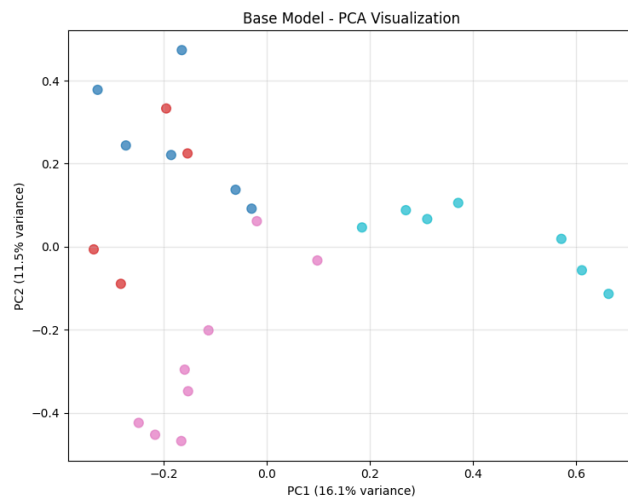
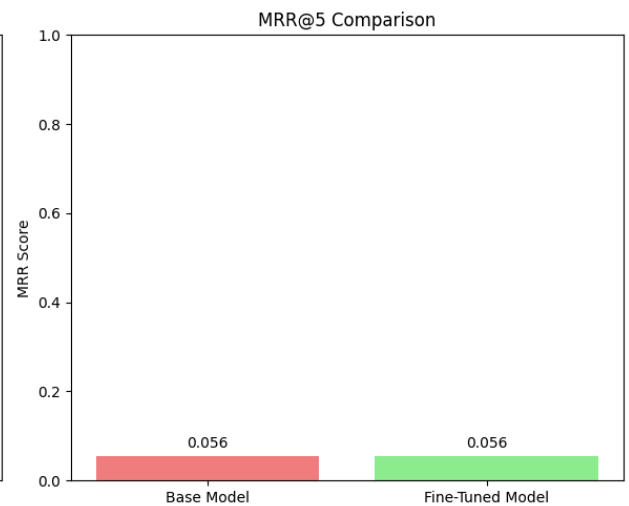
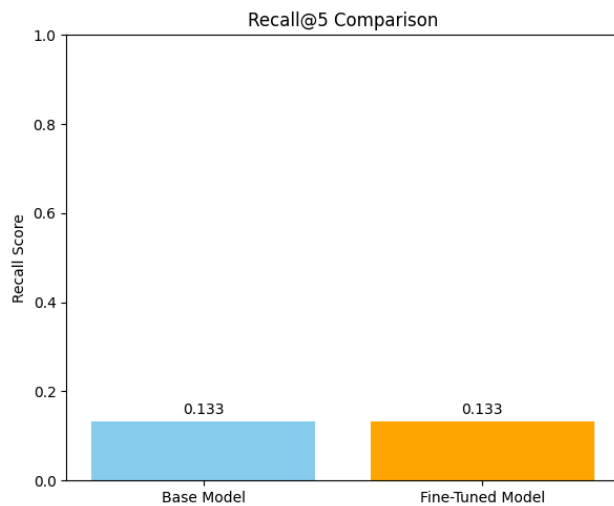
Hasil Model dan Evaluasi

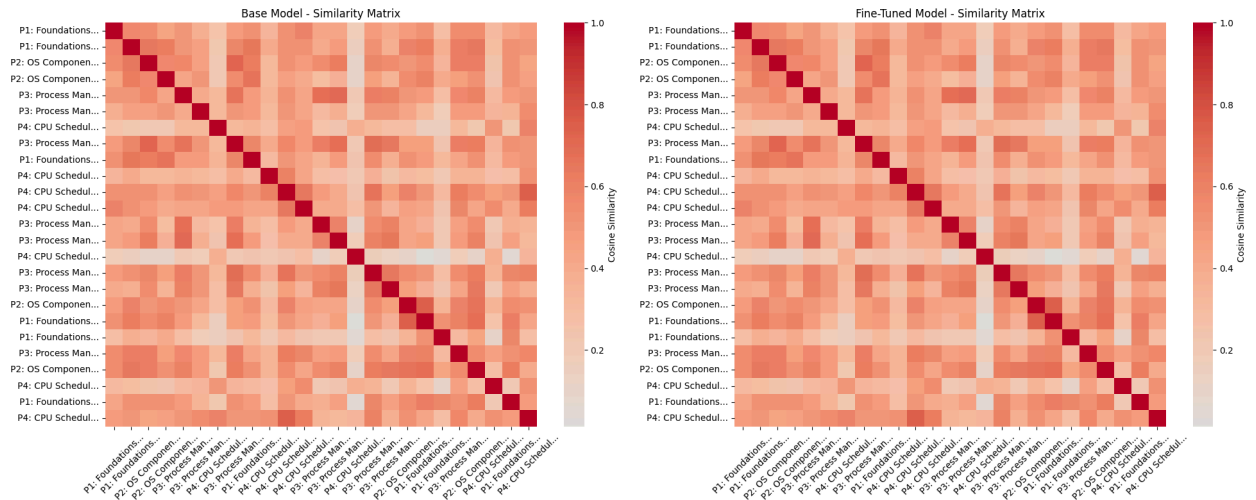
Berdasarkan laporan akhir yang Anda berikan:

- Dataset: 56 chunk dari 4 pertemuan telah diproses.
- Model: Model dasar yang digunakan adalah `all-MiniLM-L6-v2`. Model fine-tuned disimpan di

dataset/SistemOperasi/finetuned_embedding_model
_sistem_operasi.

- Hasil Evaluasi:





- Metrik performa retrieval (Recall@5 dan MRR@5) untuk model dasar dan model fine-tuned identik:
 - Base Recall@5: 0.1333
 - Base MRR@5: 0.0556
 - Finetuned Recall@5: 0.1333
 - Finetuned MRR@5: 0.0556

Interpretasi Hasil Evaluasi: Hasil evaluasi yang identik antara model dasar dan model fine-tuned menunjukkan bahwa proses fine-tuning dalam eksekusi yang menghasilkan laporan ini tidak memberikan peningkatan performa retrieval. Beberapa kemungkinan penyebabnya adalah:

1. Dataset Fine-Tuning Tidak Digunakan atau Kurang Efektif:

- Mungkin file

`embedding_finetuning_examples.jsonl` tidak ada atau kosong saat proses fine-tuning dijalankan, sehingga fine-tuning dilewati dan model "fine-tuned" sebenarnya adalah salinan dari model dasar. Log sebelumnya menunjukkan peringatan jika file ini tidak ditemukan.

- Jika dataset fine-tuning digunakan tetapi kualitas atau kuantitasnya kurang (misalnya, contohnya terlalu sedikit, kurang beragam, atau pasangan query-passage tidak cukup kuat menunjukkan relevansi), maka dampaknya pada model mungkin minimal.
2. Dataset Evaluasi Kurang Sensitif: Mungkin dataset evaluasi (`embedding_evaluation_set.jsonl`) tidak cukup sensitif untuk menangkap perbedaan kecil dalam performa, atau query dalam dataset evaluasi terlalu mudah/sulit sehingga kedua model berkinerja serupa.
 3. Parameter Fine-Tuning: Parameter seperti jumlah epoch atau batch size mungkin belum optimal untuk dataset Anda.