TUGAS BESAR 2 IF 2123 ALJABAR LINIER DAN GEOMETRI

Diajukan sebagai salah satu tugas mata kuliah Aljabar Linier dan Geometri pada Semester I Tahun Akademik 2020-2021

Kelompok abcde

oleh

Thomas Ferdinand Martin	**13519099**
Azmi Muhammad Syazwana	**13519151**
Muhammad Rayhan Ravianda	**13519201**

SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA

INSTITUT TEKNOLOGI BANDUNG

BANDUNG

2020

DAFTAR ISI

BAB I

DESKRIPSI MASALAH

1.1 Spesifikasi Tugas

Buatlah program mesin pencarian dengan sebuah website lokal sederhana. Spesifikasi program adalah sebagai berikut:

1. Program mampu menerima search query. Search query dapat berupa kata dasar maupun berimbuhan.

2. Dokumen yang akan menjadi kandidat dibebaskan formatnya dan disiapkan secara manual. Minimal terdapat 15 dokumen berbeda sebagai kandidat dokumen. Bonus: Gunakan web scraping untuk mengekstraksi dokumen dari website.

3. Hasil pencarian yang terurut berdasarkan similaritas tertinggi dari hasil teratas hingga hasil terbawah berupa judul dokumen dan kalimat pertama dari dokumen tersebut. Sertakan juga nilai similaritas tiap dokumen.

4. Program disarankan untuk melakukan pembersihan dokumen terlebih dahulu sebelum diproses dalam perhitungan cosine similarity. Pembersihan dokumen bisa meliputi hal-hal berikut ini.

    a. Stemming dan Penghapusan stopwords dari isi dokumen.

    b. Penghapusan karakter-karakter yang tidak perlu.

5. Program dibuat dalam sebuah website lokal sederhana. Dibebaskan untuk menggunakan framework pemrograman website apapun. Salah satu framework website yang bisa dimanfaatkan adalah Flask (Python), ReactJS, dan PHP.

6. Kalian dapat menambahkan fitur fungsional lain yang menunjang program yang anda buat (unsur kreativitas diperbolehkan/dianjurkan).

7. Program harus modular dan mengandung komentar yang jelas.

8. Dilarang menggunakan library cosine similarity yang sudah jadi.

BAB II
TEORI SINGKAT

2.1 Information Retrieval

2.2.1 Pengertian Information Retrieval

Information retrieval atau biasa sering disebut dengan temu kembali informasi merupakan ilmu yang bertujuan mempelajari tahap-tahapan dan metode untuk mencari kembali informasi yang tersimpan dari berbagai sumber yang akurat atau koleksi sumber informasi yang di cari atau di butuhkan. Dengan adanya tindakan indexing, panggilan , serta pemanggilan data kembali. Selain itu, information retrieval juga merupakan sebuah seni dan ilmu mencari informasi di sementara dokumen, mencari dokumen itu sendiri, mencari metadata yang menjelaskan dokumen, maupun mencari dalam database, apakah relasional database itu berdiri sendiri ataupun database hypertext jaringan seperti internet, teks , suara, gambar, atau data.

Information retrieval merupakan bagian dari *computer science* yang berkaitan dengan pengambilan informasi dari dokumen-dokumen yang didasarkan pada isi dan konteks dari dokumen itu sendiri. Tujuan dari information retrieval, yaitu untuk memenuhi informasi pengguna dengan cara meretrieve dokumen yang akurat atau mengurangi dokumen pencarian yang tidak relevan atau akurat.

Secara konsep yang sederhana dari sistem temu balik ialah bentuk proses untuk mencari, dan menemukan informasi yang di cari dan apabila di titik beratkan terhadap prosesnya maka di dalamnya terungkap bagaimana bentuk dari perjalanan informasi yang di minta, menjadi informasi yang di berikan, menrut para ahli IR di definisikan sebagai berikut:

- Kowalski
  Information retrieval merupakan konsep yang sederhana dalam melakukan pencarian yang dilakukan seseorang. Sebagai contoh ketika pengguna akan mencari informasi yang di inginkan maka sistem menerjemahkan kepada bentuk statement yang kemudian di eksekusi oleh sistem pencari tersebut.

- William hersh

  IR merupakan bentuk ilmu sistem informasi dan ilmu komputerisasi yang merujuk ke pengurutan dan pengambilan bentuk informasi yang heterogen dan sebagian besar-tekstil.

- Kutipan wikipedia

  Temu balik informasi merupakan ilmu untuk mencari informasi serta mencari dokumen tersebut pada database.

Berdasarkan dari ketiga ahli tersebut dapat disimpulkan IR merupakan ilmu dalam teknologi informasi yang menjabarkan tentang proses pencarian dan pengambilan kembali informasi. Jadi, IR atau disebut temu balik informasi adalah sebuah istilah yang generic dan mengacu pada temu balik informasi dokumen atau sumber data dari beberapa fakta yang di miliki unit informasi lain atau perpustakaan.

2.2.2 Konsep Dasar Information Retrieval

- Indexing

Merupakan cara pengindeksan dokumen yang mencakup proses pencatatan dari ciri-ciri dokumen, analisa isi, klasifikasi maupun pembuatan list entri ke informasi tersebut. Tujuan indexing ialah agar mempermudah ditemukannya dokumen yang relevan dengan pertanyaan dengan akurat dan tepat.

- Searching

Merupakan proses untuk mencari dan menemukan kembali dokumen yang relevan dengan query yang di masukan.

- Perengkingan Relevansi Keyword Query

Merupakan proses pencarian data berdasarkan peringkat dokumen yang lebih banyak di gunakan dan sesuai dengan query yang di inputkan dan infromasi akan di tampilkan berdasarkan tingkatan dari dokumen.

2.2.3 Metode-Metode Information Retrieval

- Inverted Index

Inverted index adalah sebuah struktur data index yang dibangun untuk memudahkan query

pencarian yang memotong tiap kata (term) yang berbeda dari suatu daftar term dokumen. Inverted index memiliki tujuan untuk meningkatkan kecepatan dan efisiensi dalam melakukan pencarian pada sekumpulan dokumen dan menemukan dokumen-dokumen yang mengandung query user (CatenaCraig, Macdonald, & Ounis, 2014).

Information need merupakan topik dimana user ingin tahu lebih jauh, sedangkan query merupakan cara user berkomunikasi dengan komputer untuk memperoleh informasi yang diinginkan. Oleh karena itu, agar hasil yang diperoleh memiliki hasil yang baik dilakukan pengujian pada sistem IR. Pengujian efektifitas system IR menggunakan dua acara yaitu precision dan recall. Precision adalah tingkat ketepatan antara informasi yang diminta oleh pengguna dengan jawaban yang diberikan oleh sistem. Sedangkan recall adalah tingkat keberhasilan sistem dalam menemukan kembali sebuah informasi (Bucher, Clarke, & Cormack, 2010). Adapun langkah-langkah yang dilakukan pada inverted index antara lain:

1. Menentukan dokumen yang akan diindeks;
2. Melakukan Tokenize teks, tiap dokumen menjadi token;
3. Membuat dictionary dan posting list;
4. Melakukan preprocessing linguistic dan menghasilkan token;
5. Mengindeks dokumen dimana tiap term terjadi dengan membuat inverted index.

Setiap dokumen memiliki serial number unik untuk dikenal sebagai identitas dokumen. Pada proses konstruksi indeks, kit adapt memberkan nilai trigger untuk tiap dokumen baru yang ditemukan. Selanjutnya adalah mengurutkan sehingga term alfabetis. Dictionary merekam statiskik seperti jumlah dokumen yang berisi tiap term yang berguna pada search engine pada saat query.

- Boolean Retrieval

Boolean Retrieval merupakan proses pencarian informasi dari query yang menggunakan ekspresi Boolean (Bucher et al., 2010). Dengan ekspresi boolean dengan menggunakan operator logika AND, OR dan NOT. Dalam menentukan hasil perhitungannya hanya berupa nilai binary (1 atau 0). Hasil boolean retrieval yang ada hanya dokumen relevan atau tidak sama sekali. Sehingga keunggulan dari boolean retrieval tidak menghasilkan dokumen yang sama.

Dalam pengerjaan operator boolean (AND, NOT, OR) ada urutan pengerjaannya (operator precedence). Dalam implementasinya akan memprioritaskan yang berada dalam kurung (), baru selanjutnya NOT, AND, dan OR. Boolean retrival melakukan perbaikan karena datanya terlalu besar bila tersimpan dalam komputer, seperti ini kita perlu memenuhi peraturannya diantaranya kecepatan dalam pemrosesan dokumen yang sangat banyak, fleksibilitas dan perangkingan.

- Tokenization

Tokenization adalah metode pemecah teks menjadi token-token yang berurutan. Proses tokenization primitif biasanya hanya memecah teks dengan whitespace sebagai pembagi, lalu mengubahnya menjadi huruf kecil supaya seragam.

Tokenisasi secara garis besar memecah sekumpulan karakter dalam suatu teks ke dalam satuan kata, bagaimana membedakan karakter-karakter tertentu yang dapat diperlakukan sebagai pemisah kata atau bukan. Sebagai contoh karakter whitespace, seperti enter, tabulasi, spasi dianggap sebagai pemisah kata. Namun untuk karakter petik tunggal (‘), titik (.), semikolon (;), titk dua (:) atau lainnya, dapat memiliki peran yang cukup banyak sebagai pemisah kata.
Stemming adalah proses untuk mendapatkan kata dasar dengan cara menghapus imbuhan kata. Proses stem adalah bagian dari pra-pemrosesan dokumen teks dalam pencarian informasi. Stemming merupakan sebuah proses yang bertujuan untuk mereduksi jumlah variasi dalam representasi dari sebuah kata (Kowalski, 2011). Resiko dari proses stemming adalah hilangnya informasi dari kata yang di-stem. Hal ini menghasilkan menurunnya akurasi atau presisi. Sedangkan untuk keuntungannya adalah, proses stemming bisa meningkatkan kemampuan untuk melakukan recall.

- Stemming and Lemmatization

Tujuan dari stemming adalah untuk meningkatkan akurasi pencarian teks (D. Sharma, 2012). untuk meningkatkan performace dan mengurangi penggunakan resource dari sistem dengan mengurangi jumlah unique word yang harus diakomodasikan oleh sistem. Stemming juga diperlukan dalam mengompresi algoritma teks (Sinaga, Adiwijaya, & Nugroho, 2015). Jadi, secara umum, algoritma stemming mengerjakan transformasi dari sebuah kata menjadi sebuah standar representasi morfologi (yang dikenal sebagai stem).

Dalam stemming bahasa Inggris, algoritma Porter stemming adalah algoritma yang lebih sederhana dari algoritma stemming sebelumnya dari stemmer tipe Lovin. Algoritma Porter berkurang kompleksitas aturan dalam penghapusan suffix. Oleh karena itu, algoritma ini menjadi standar untuk stemmer dan menyediakan bahasa Inggris model untuk pemrosesan bahasa lain (Willett, 2006).

Dalam stemming Bahasa Indonesia, ada beberapa algoritma stemming Bahasa Indonesia antara lain algoritma Nazied dan Andriani, algoritma dari Arifin dan Setiono, algoritma Vega, algoritma William, dan Tahaghoghi, dan algoritma Arifin, Mahendra, Ciptaningtyas.

Lemmatization adalah proses menemukan bentuk dasar dari sebuah kata (Ingason, Helgadóttir, Loftsson, & Rögnvaldsson, 2008). Lemmatization adalah proses normalisasi pada teks/kata dengan berdasarkan pada bentuk dasar yang merupakan bentuk lemma-nya. Normalisasi adalah mengidentifikasikan dan menghapus prefiks serta suffiks dari sebuah kata. Lemma adalah bentuk dasar dari sebuah kata yang memiliki arti tertentu berdasar pada kamus.

- Vector Space Model

Vector space model (VSM) adalah teknik dasar dalam perolehan informasi yang dapat digunakan untuk penilaian relevansi dokumen terhadap kata kunci pencarian (query) pada mesin pencari, klasifikasi dokumen, dan pengelompokan dokumen (Adriani, M., Asian, J., Nazief, B., & et al.,2007). Vector space model merupakan representasi kumpulan dokumen sebagai vektor dalam sebuah ruang vector (Akerkar, R., 2005). Dalam Vector Space Model, koleksi dokumen direpresentasikan sebagai sebuah matrik term-document (matrik term-frequency). Setiap sel dalam matrik bersesuaian dengan bobot yang diberikan dari suatu term dalam dokmen yang ditentukan. Nilai nol berarti bahwa term tersebut tidak hadir dalam dokumen .

Melalui vector space model dan TF weighting maka akan didapatkan representasi nilai numerik dokummen sehingga kemudian dapat dihitung kedekatan antar dokumen. Semakin dekat dua vektor di dalam suatu VSM, maka semakin mirip dua dokumen yang diwakili vektor tersebut. Fungsi untuk mengukur kemiripan (similarity measure) yang dapat digunakan untuk model ini terdiri dari:

1. Cosine distance / cosine similarity
2. Inner similarity
3. Dice similarity

4. Jaccard similarity

2.2.4 Elemen Penting Information Retrieval

Ada dua elemen penting dalam information retrieval, yaitu Precision dan Recall.

Precision merupakan rasio jumlah dokumen relevan yang ditemukan dengan total jumlah dokumen yang ditemukan oleh SE. Precision mencerminkan kualitas himpunan jawaban, tetapi tidak memandang total jumlah dokumen yang relevan seraya kumpulan dokumen.

$$Precision = \frac{|\{relevant\ documents\} \cap \{documents\ retrieved\}|}{|\{documents\ retrieved\}|}$$

Recall merupakan rasio jumlah dokumen relevan yang ditemukan kembali dengan total jumlah dokumen seraya kumpulan dokumen yang dianggap relevan.

$$Recall = \frac{|\{relevant\ documents\} \cap \{documents\ retrieved\}|}{|\{relevant\ documents\}|}$$

2.2 Vector dan Cosine Similarity

Secara umum, fungsi similarity adalah fungsi yang menerima dua buah objek dan mengembalikan nilai kemiripan (similarity) antara kedua objek tersebut berupa bilangan riil. Umumnya, nilai yang dihasilkan oleh fungsi similarity berkisar pada interval [0...1]. Namun ada juga beberapa fungsi similarity yang menghasilkan nilai yang berada di luar interval tersebut. Untuk memetakan hasil fungsi tersebut pada interval [0...1] dapat dilakukan normalisasi [1].

Cosine similarity adalah perhitungan kesamaan antara dua vektor n dimensi dengan mencari kosinus dari sudut diantara keduanya dan sering digunakan untuk membandingkan dokumen dalam text mining [8]. Rumus Cosine similarity adalah sebagai berikut:

$$sim(\mathbf{Q}, \mathbf{D}) = \cos\theta = \frac{\mathbf{Q} \cdot \mathbf{D}}{\|\mathbf{Q}\|\|\mathbf{D}\|}$$

Q, D : *vector dot product* dari Q dan Y

|| Q || : panjang vektor Q

|| D || : panjang vektor D

Pang-Ning Tan menjelaskan bahwa semakin besar hasil fungsi similarity, maka kedua objek yang dievaluasi dianggap semakin mirip. Jika sebaliknya, maka semakin kecil hasil fungsi similarity, maka kedua objek tersebut dianggap semakin berbeda. Pada fungsi yang menghasilkan nilai pada jangkauan [0...1], nilai 1 melambangkan kedua objek persis sama, sedangkan nilai 0 melambangkan kedua objek sama sekali berbeda.

BAB III
IMPLEMENTASI PROGRAM

3.1 Class

Terdapat 3 class yang didefinisikan, yaitu:

- class Dokumen
- class CustomDf
- class Query

a. class Dokumen

Pada class Dokumen terdapat 7 instance attribute yang digunakan yaitu

- filename : menyatakan judul dokumen
- stem : menyatakan token dokumen
- count_words : menyatakan jumlah term dalam dokumen
- first : menyatakan kalimat pertama dalam dokumen
- data_info : menyatakan info-info dokumen dalam bentuk dict, info yang terkandung adalah kumpulan dari instance attribute
- vektor : menyatakan vektor jumlah term
- cos_sim : menyatakan similaritas dokumen dengan query

Terdapat 7 method pada class ini

- constructor : parameter nama file
- ProsesFile : melakukan pembacaan file, pembersihan file, pembentukan token, pencarian info dokumen seperti jumlah term dan kalimat pertama
- Name, Vektor, cosSim, firstSentence: mengembalikan instance attribute
- dokumenInfo: mengembalikan instance attribute dari data_info

b. class CustomDf

class ini dibuat untuk mempermudah pemrosesan data frame yang digunakan, hanya terdapat satu instance attribute yang digunakan yaitu df yang merupakan data frame

Method dalam class ini berupa:

- openDf : membuka data frame dari lokasi yang ditentukan

- createDf : membuat data frame baru dengan parameter kolom baru
- insertRowDf : menambahkan data pada baris baru
- save : menyimpan data frame dengan format csv pada lokasi yang ditentukan
- updateDf : mengubah attribute df dengan data frame lain
- getDf : mengembalikan data frame dalam class

c. class Query

Pada class ini terdapat 4 atribut, yaitu:

- query : merupakan query yang diterima
- vektor_query : merupakan token dari query
- count_query : merupakan jumlah term yang muncul dalam query
- count_words_query : merupakan vektor jumlah term query yang muncul pada dokumen yang sudah di upload
- dfTerm : merupakan data frame yang berisi jumlah term query di masing-masing dokumen

Kemudian terdapat 5 method dan 1 constructor, method tersebut antara lain:

- termQuery : memproses query untuk mendapatkan token query, jumlah term pada query
- countQuery : memproses query untuk mendapat vektor query
- cosSim : memproses query dan dokumen, untuk mendapatkan tingkat kemiripan berdasarkan query
- getTerm : mengembalikan token query
- getVektor : mengembalikan jumlah token query

3.2 Garis besar program

Secara garis besar langkah-langkah pemrosesan file hingga pemerolehan tingkat kemiripan dijabarkan dalam urutan berikut:

1. Web menerima file, namun sebelum disimpan, nama file diubah menjadi nama file yang aman dan dipastikan agar tidak terdapat dokumen yang dobel.
2. Setelah semua file di upload, dokumen dibaca satu per satu.
3. Proses pertama, dilakukkan penghapusan karakter yang tidak dibutuhkan memanfaatkan library re, lalu dilakukan *stemming* dan penghapusan *stopwords* pada program dengan

memanfaatkan library Sastrawi (karena file yang di-upload adalah berbahasa Indonesia). Data stopwords diperoleh dari library Sastrawi sehingga mungkin tidak terlalu update dengan *stopwords* yang ada sekarang.

4. Proses kedua, dilakukan pembentukan vektor yang merepresentasikan jumlah kemunculan term pada dokumen, dan penambahan term yang muncul pada seluruh dokumen, sehingga diperoleh 2 vektor yaitu tabel term dan vektor jumlah term yang indeksnya saling berkorespondensi, artinya misalkan baris 1 tabel term adalah ‘term1’ maka indeks pertama dari vektor jumlah term menyatakan jumlah kemunculan ‘term1’ pada dokumen terkait.
5. Proses ketiga, dilakukan pencarian info dokumen seperti kalimat pertama, jumlah term, dan judul dokumen
6. Proses keempat, info dokumen yang telah diperoleh, ditambahkan ke dalam data frame yang berisi info-info dari seluruh dokumen yang di-upload.
7. Setelah itu, pengguna memasukkan query dan query akan diproses.
8. Proses pertama, akan dilakukan penghapusan simbol tidak penting, perlakuan *stemming* dan penghapusan *stopwords,* pengubahan query menjadi token, dan perhitungan jumlah kemunculan query sama seperti perlakuan pada dokumen.
9. Proses kedua, setelah vektor jumlah term pada query didapat, akan dicari kemiripan dokumen dengan perhitungan cosine similarity. Program akan menerima data frame berisi info dokumen yang telah di-upload, kemudian dilakukan iterasi untuk mencari tingkat kemiripan masing-masing dokumen. Setelah itu, dicari jumlah kemunculan term pada masing-masing dokumen, data kemunculan term ditambahkan ke data frame lain
10. Proses ketiga, dokumen disortir dari tingkat kemiripan tertinggi, dan tidak akan ditampilkan apabila tingkat kemiripan adalah 0 dan panjang query adalah 0.

3.3 Fungsi

Fungsi yang digunakan, ditulis dalam source code lain, fungsi-fungsi tersebut antara lain:

- stemText : melakukan pembersihan pada dokumen seperti stemming, penghapusan stopwords dan penghapusan simbol tidak penting
- readtextToString : membuka dan membaca file ektensi txt
- token : mengubah string menjadi token
- dotProduct : menghitung dot product
- distanceVektor : menghitung magnitude vektor

- cosineSimilarity : menghitung nilai cos dari 2 vektor
- stemm : melakukan pembersihan dokumen, pembuatan token dokumen dengan ekstensi baik html maupun txt
- stemm2 : sama seperti stemm namun dengan parameter string
- first_sentence : mencari kalimat pertama pada dokumen
- stringToInt : mengubah array of integer yang disimpan sebagai string, menjadi array of integr (pemrosesan pada csv)
- htmlParser : membaca isi dokumen html

3.4 Library Pemrosesan File

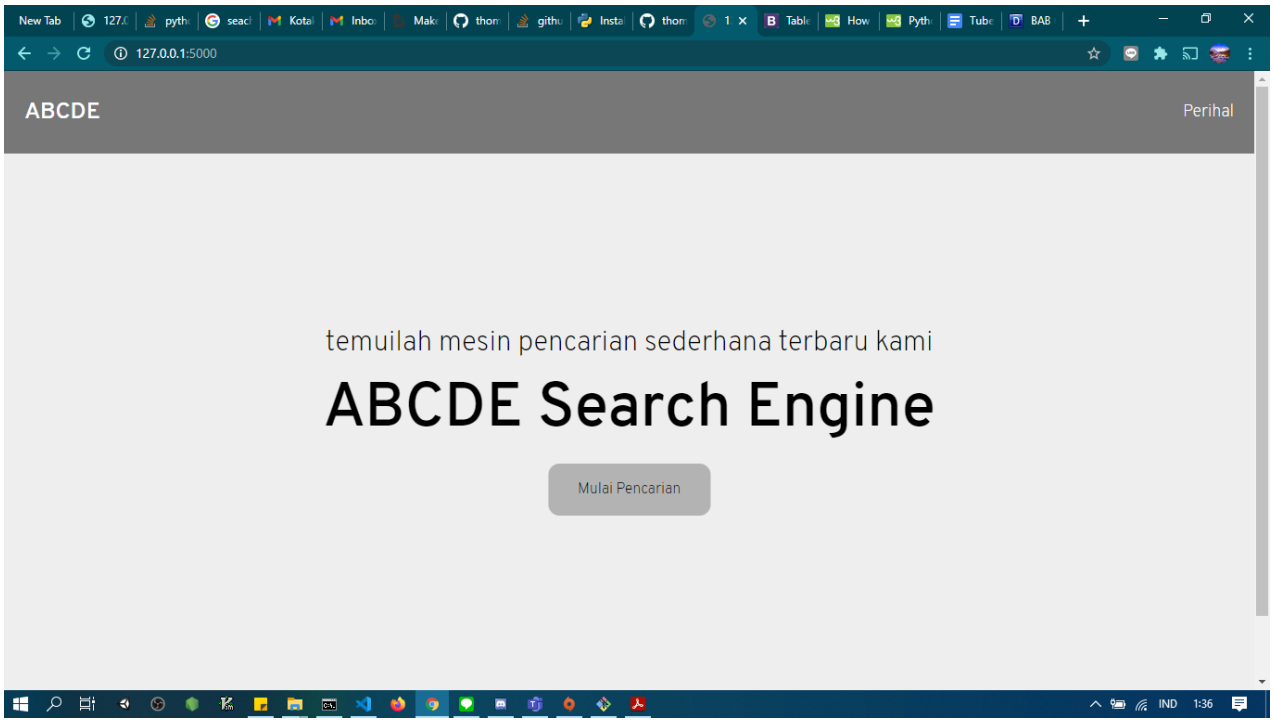
Library yang digunakan untuk memproses file:

- Sastrawi : stemming dan penghapusan stopwords
- re : penghapusan simbol tidak penting
- BeautifulSoup : pembacaan teks pada html
- pandas : pembentukan dataframe

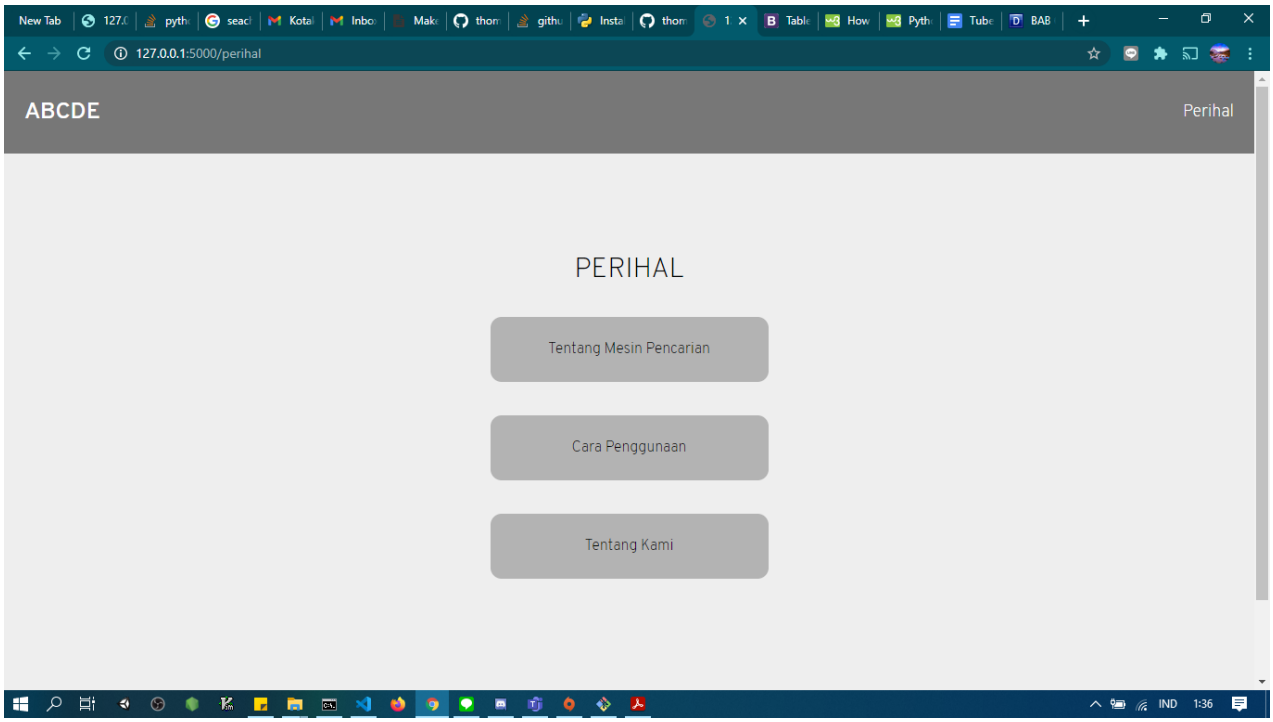
BAB IV
EKSPERIMEN

4.1 Analisis Program

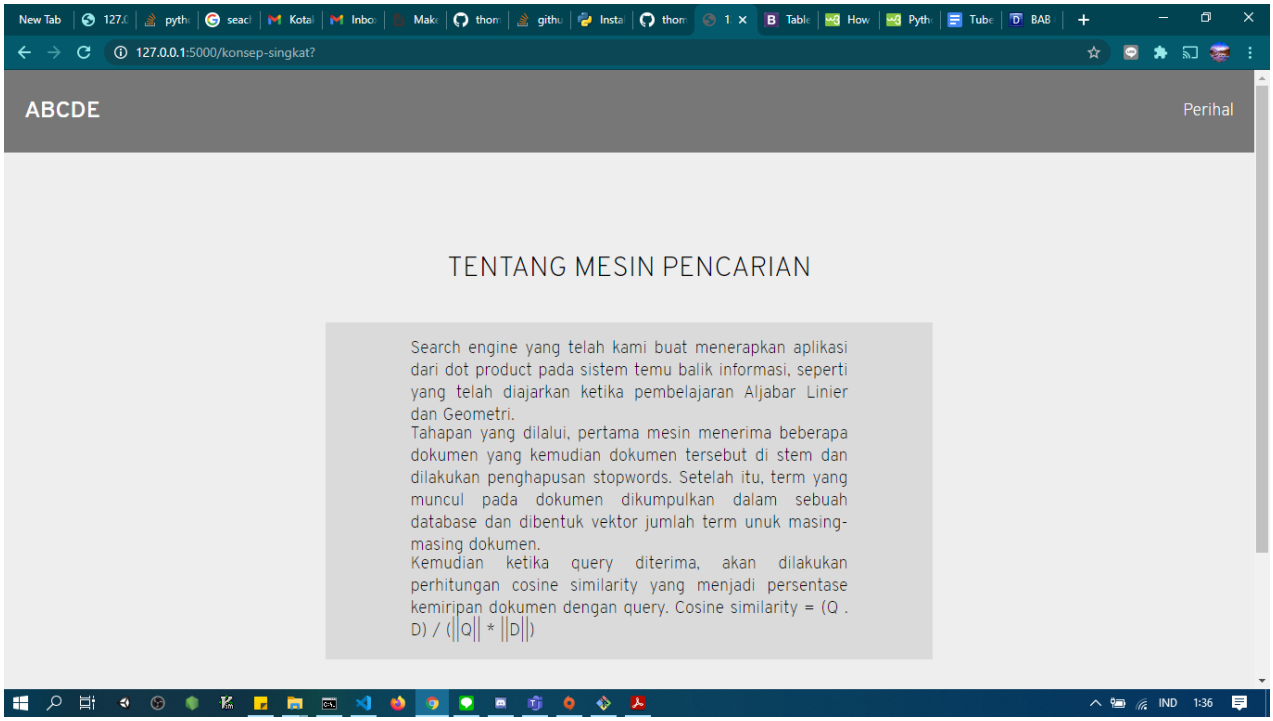
1. Halaman utama



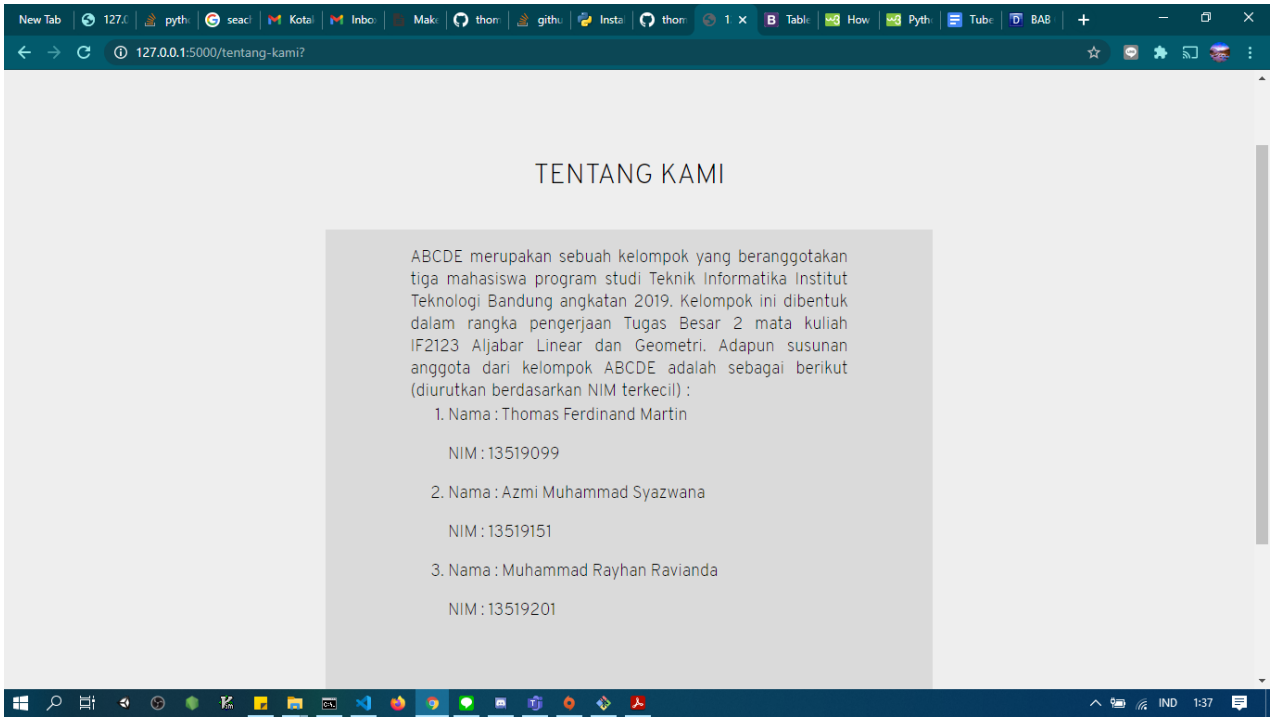
2. Halaman perihal



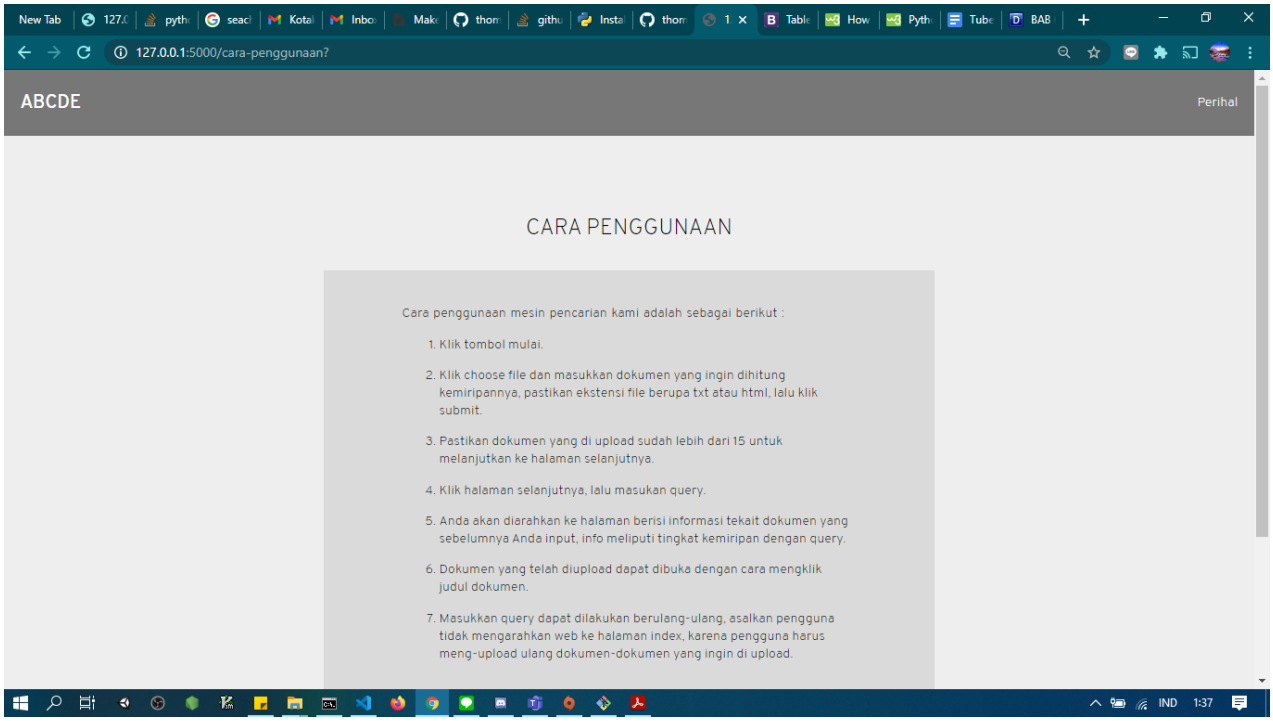
3. Halaman mengenai konsep singkat



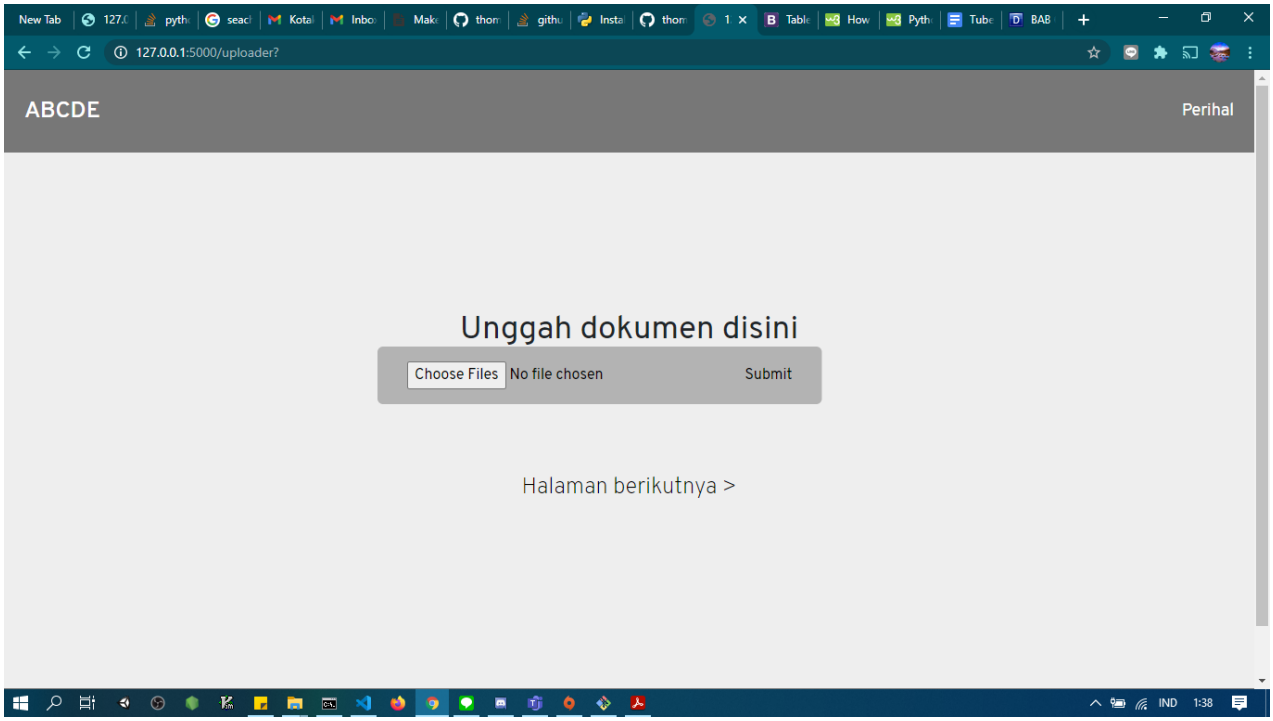
4. Halaman about us



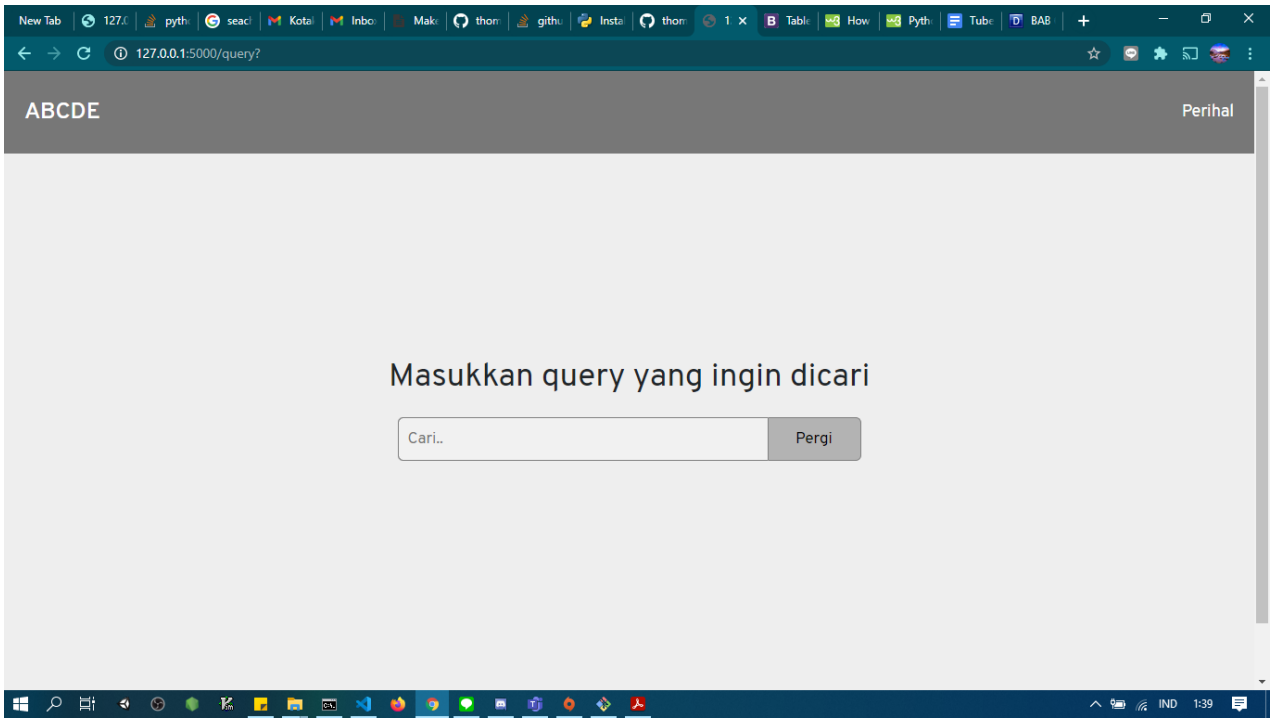
5. Halaman cara penggunaan



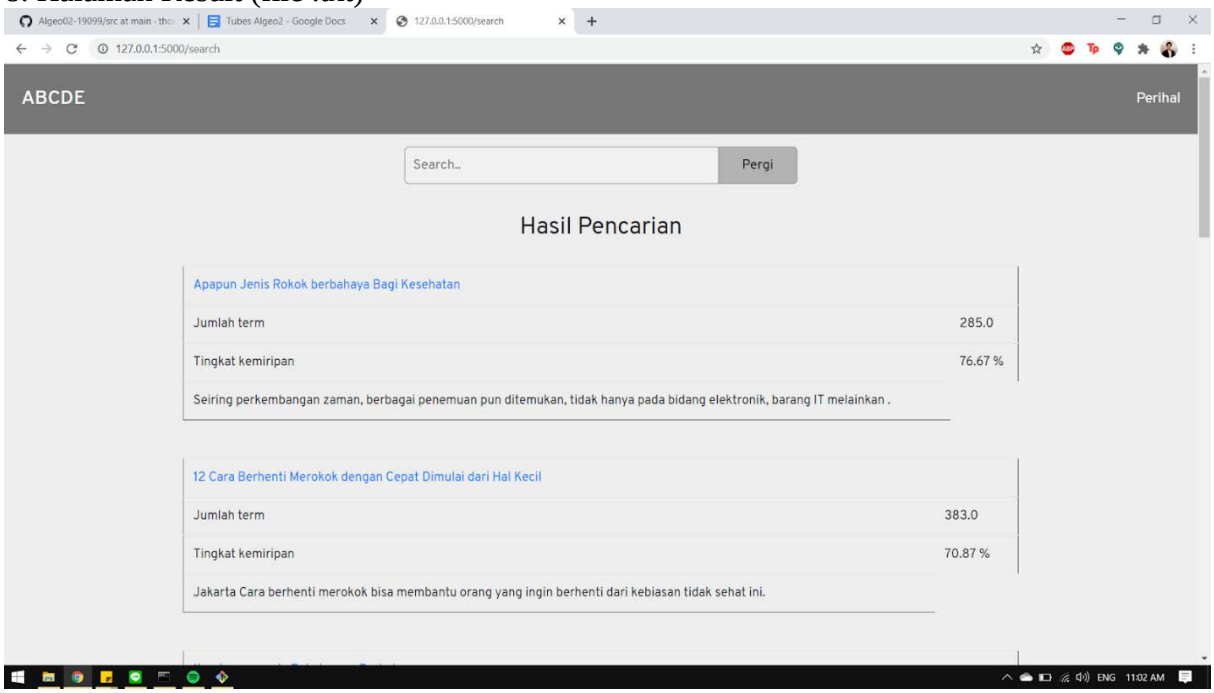
6. Halaman uploader

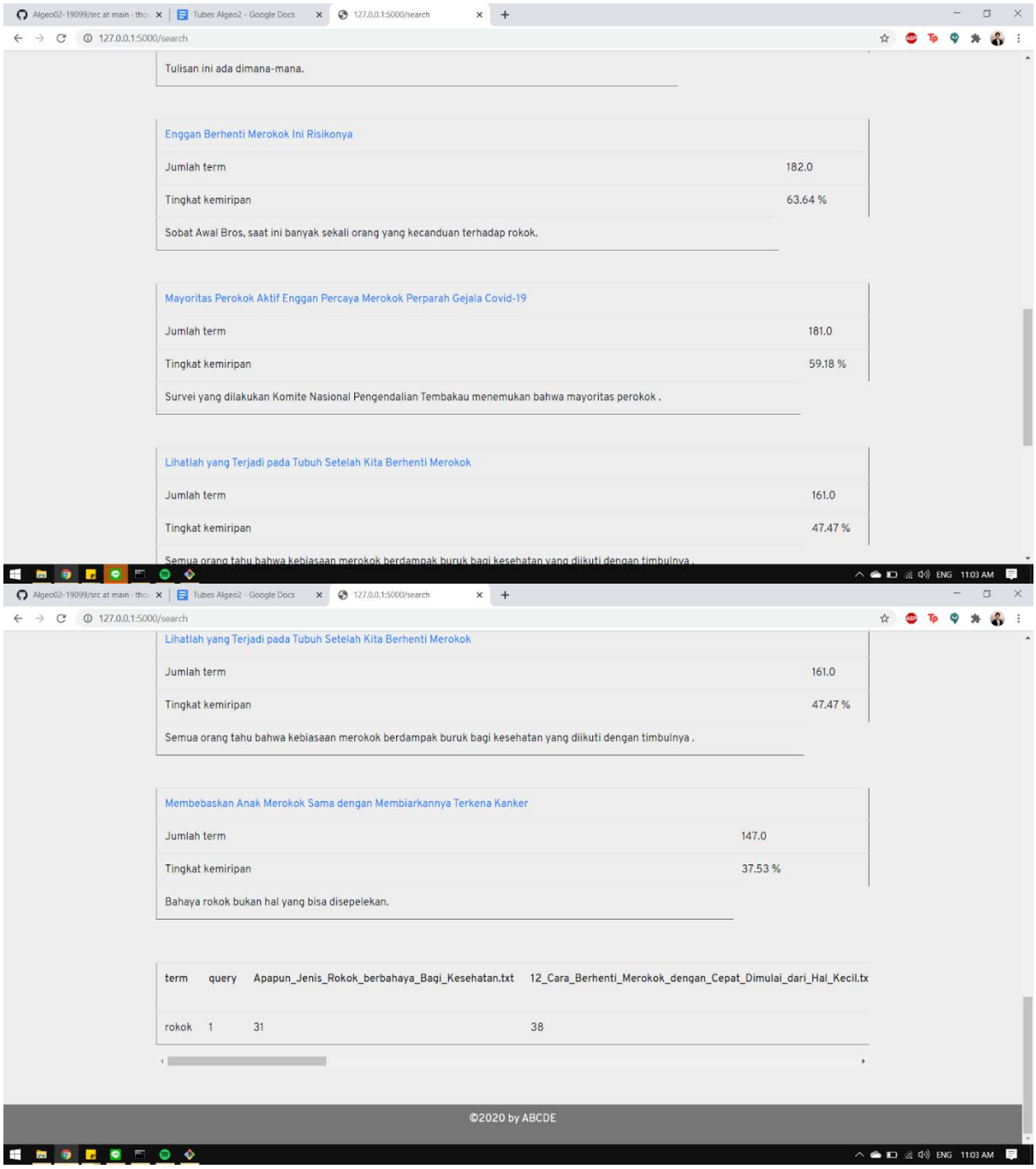


7. Halaman query

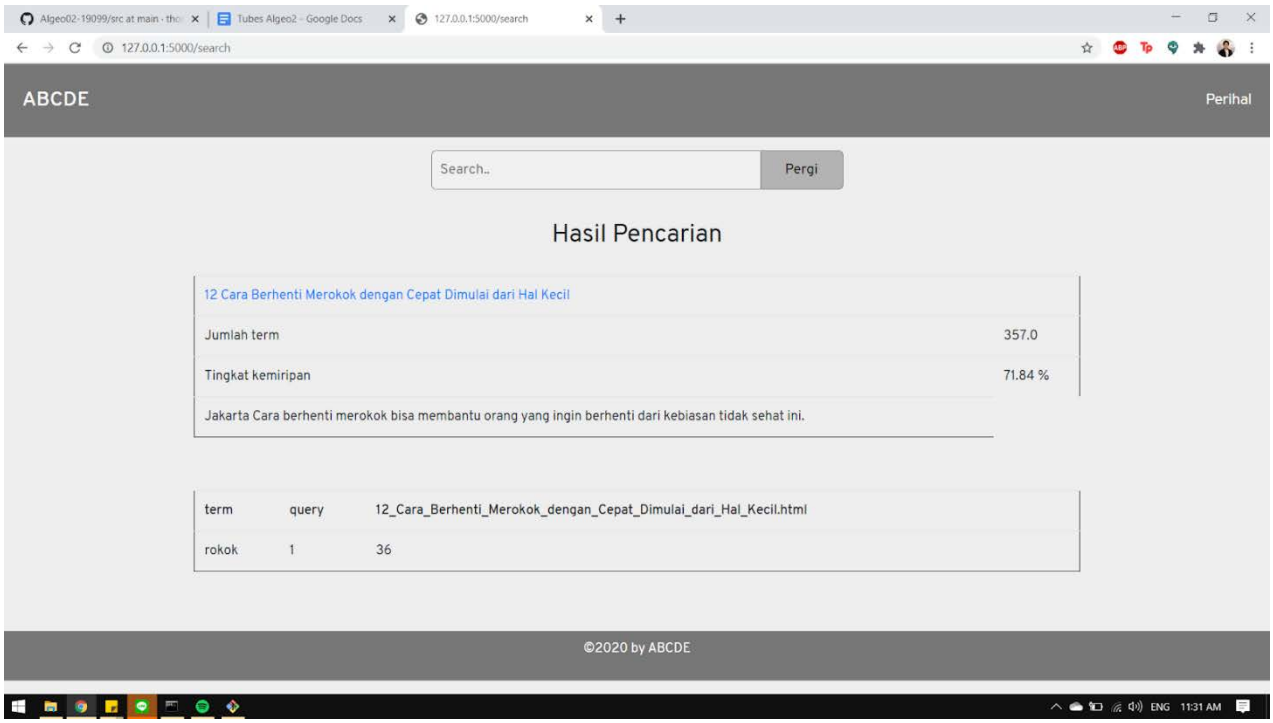


8. Halaman Result (file .txt)

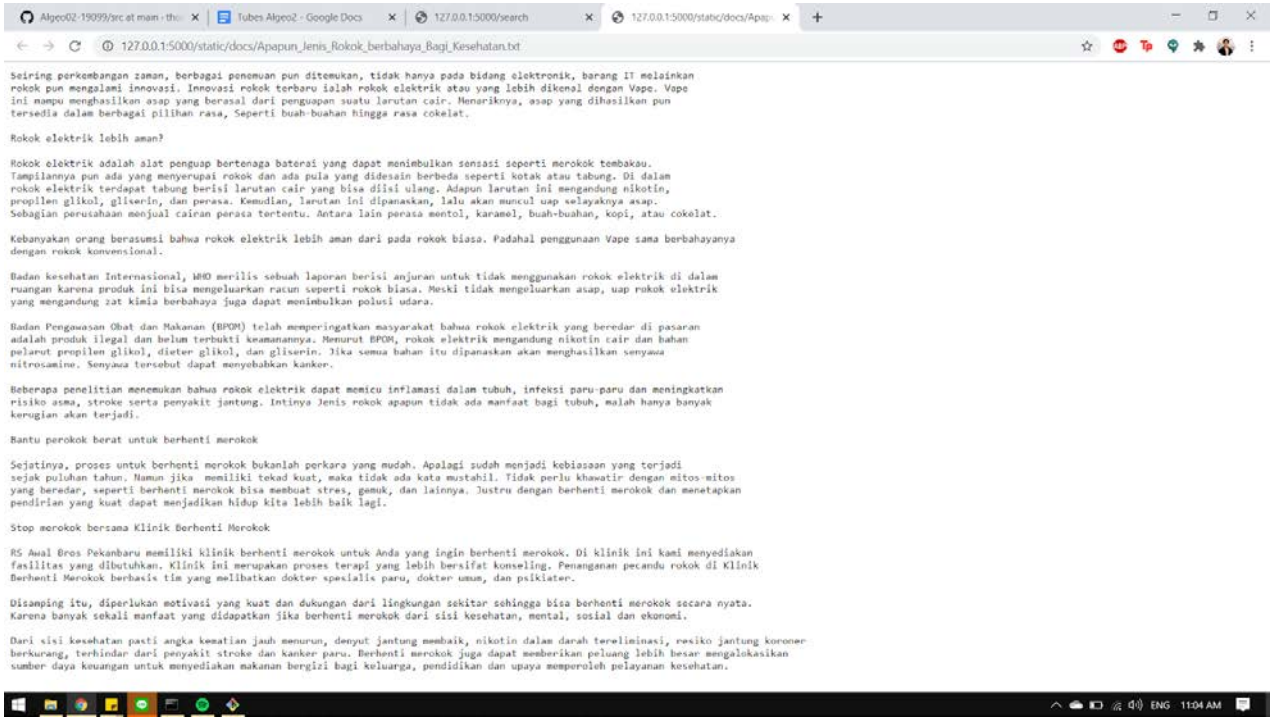



Algeo02-19099/src at main - thor
Tubes Algeo2 - Google Docs
127.0.0.1:5000/search
127.0.0.1:5000/search
Tulisan ini ada dimana-mana.
Enggan Berhenti Merokok Ini Risikonya
Jumlah term 182.0
Tingkat kemiripan 63.64 %
Sobat Awal Bros, saat ini banyak sekali orang yang kecanduan terhadap rokok.
Mayoritas Perokok Aktif Enggan Percaya Merokok Perparah Gejala Covid-19
Jumlah term 181.0
Tingkat kemiripan 59.18 %
Survei yang dilakukan Komite Nasional Pengendalian Tembakau menemukan bahwa mayoritas perokok .
Lihatlah yang Terjadi pada Tubuh Setelah Kita Berhenti Merokok
Jumlah term 161.0
Tingkat kemiripan 47.47 %
Semua orang tahu bahwa kebiasaan merokok berdampak buruk bagi kesehatan yang diikuti dengan timbulnya .
ENG 11:03 AM
Algeo02-19099/src at main - thor
Tubes Algeo2 - Google Docs
127.0.0.1:5000/search
127.0.0.1:5000/search
Lihatlah yang Terjadi pada Tubuh Setelah Kita Berhenti Merokok
Jumlah term 161.0
Tingkat kemiripan 47.47 %
Semua orang tahu bahwa kebiasaan merokok berdampak buruk bagi kesehatan yang diikuti dengan timbulnya .
Membebaskan Anak Merokok Sama dengan Membiarkannya Terkena Kanker
Jumlah term 147.0
Tingkat kemiripan 37.53 %
Bahaya rokok bukan hal yang bisa disepelekan.
term query Apapun_Jenis_Rokok_berbahaya_Bagi_Kesehatan.txt 12_Cara_Berhenti_Merokok_dengan_Cepat_Dimulai_dari_Hal_Kecil.tx
rokok 1 31 38
©2020 by ABCDE
ENG 11:03 AM

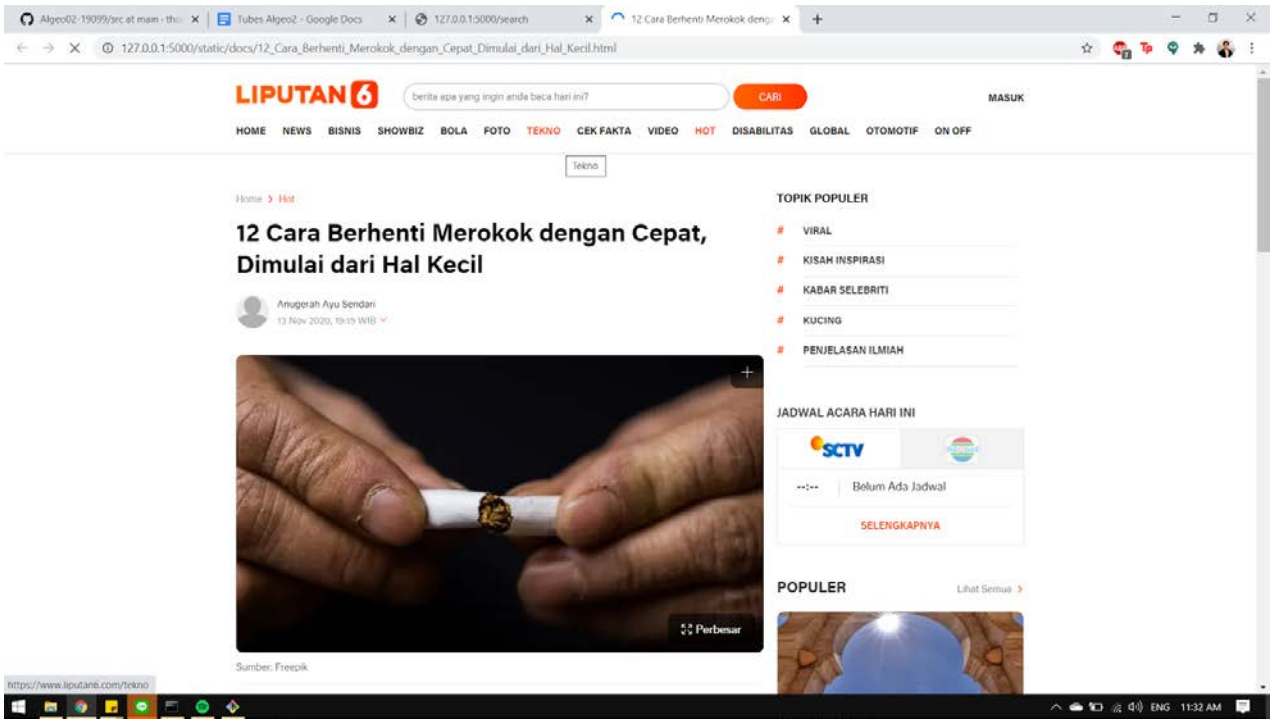
9. Halaman Result (file .html)



10. Halaman File (file .txt)



11. Halaman File (file .html)



Web menampilkan daftar kemiripan dokumen setelah query di terima dan mengurutkan dari tingkat kemiripan dalam persen. Dalam hasil ranking diperoleh jumlah term dalam float, kami kurang tahu kenapa bisa seperti itu padahal sudah di casting ke integer dan panjang array harusnya selalu integer.

Jumlah term dari token query pada masing-masing dokumen ditampilkan pada tabel paling bawah dan diurutkan dari dokumen dengan kemiripan tertinggi dari kiri ke kanan. Proses upload dokummen terjadi cukup lama sekitar 4-5 menit karena proses backend kami yang menggunakan python, dimana python sendiri kurang powerful untuk masalah waktu proses

Ketika dokumen di-upload, terjadi proses stemming, penghapusan stopwords, perubahan menjadi token, dan pembentukan vektor jumlah term untuk masing-masing dokumen, sehingga ketika di halaman search, web hanya tinggal menghitung kemiripan dengan menggunakan cosine similarity. Pembagian tugas ini untuk mencegah kalkulasi yang terlalu lama di satu halaman dan menyebabkan server overload. Hasil web scrapping dan perhitungan nilai kemiripan masih kurang maksimal karena informasi yang berhasil di retrieve terbatas pada paragraf dengan asumsi sebuah paragraf terdiri dari minimal 3 kalimat.

BAB V
KESIMPULAN, SARAN, DAN REFLEKSI

5.1 Kesimpulan

Vektor yang kita pelajari memiliki banyak aplikasi dalam kehidupan, terutama dunia pemrograman dan internet seperti machine learning dan mesin pencarian dokumen seperti google. Pada tugas ini, kami membuat mesin pencarian dokumen dengan perhitungan cosine similarity dari vektor jumlah kemunculan term pada dokumen dan query. Dokumen yang ingin dihitung kemiripannya akan di-*stemming* terlebih dahulu, yaitu pengubahan kata pada dokumen menjadi kata dasar, kemudian penghapusan *stopwords* yang merupakan kata sambung, kata hubung dan sebagainya, dan penghapusan simbol-simbol tidak penting. Pengubahan isi dokumen ini dilakukan untuk mendapatkan tingkat kemiripan yang semakin akurat.

5.2 Saran

Sebaiknya deadline mungkin bisa lebih diperpanjang lagi agar kami dapat lebih baik lagi dalam mempelajari dan mendalami mengenai web development. Walaupun begitu, tugas besar ini sangat membantu kami untuk mulai mempelajari dan mendalami web development yang sangat dibutuhkan pada saat ini.

5.3 Refleksi

Terdapat banyak hal yang dipelajari dari tubes kali ini terutama dalam bidang web development. Tubes ini membuat kami belajar mengenai API, backend, frontend dan segala hal web-dev lainnya. Proses terlama dan memakan waktu dalam pengerjaan tubes ini adalah eksplorasi mengenai tools, framework, dan bahasa pemrograman yang paling cocok untuk digunakan.

DAFTAR PUSTAKA

- Adriani, M., Asian, J., Nazief, B., & et al. (2007). Stemming Indonesian: A Confix-Stripping Approach. ACM Transactions on Asian Language Information Processing, 6, 1–33.
- https://id.wikipedia.org/wiki/Sistem_temu_balik_informasi
- Kowalski, M. (2011). Information Retrieval Architecture and Algorithms. New York: Springer.
- Bucher, S., Clarke, C. L., & Cormack, G. V. (2010). Information Retrieval – Implementing and Evaluation Search Engines. MIT Press.
- https://garudacyber.co.id/artikel/1436-pengertian-dan-konsep-information-retrieval
- https://onlinelearning.binus.ac.id/computer-science/post/metode-metode-information-retrieval/