

LAPORAN TBO TUGAS PROYEK 1
“Finite Automata Application in Search Engine”



Ditujukan sebagai tugas mata kuliah Teori Bahasa dan Otomata prodi Informatika
Universitas Udayana
(Dosen : **Dr. Anak Agung Istri Ngurah Eka Karyawati, S.Si., M.Eng.**)

Kelompok 4 :

Ni Made Yuli Cahyani	1808561027	B
I Made Prema Satwika	1808561031	B
Ni Made Ayu Suandewi	1808561036	B
Made Dwiki Budi Laksana	1808561041	B
I Gusti Gede Ngurah Adryan Pratama Putra Bueka	1808561045	B

PROGRAM STUDI INFORMATIKA
FAKULAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS UDAYANA
2020

KATA PENGANTAR

Puji dan syukur ke hadirat Tuhan Yang Maha Kuasa atas segala rahmat yang diberikan-Nya sehingga Laporan TBO Tugas Proyek 1 ini dapat kami selesaikan. Laporan ini kami buat sebagai kewajiban untuk memenuhi tugas. Dalam kesempatan ini, penulis menghaturkan terimakasih yang dalam kepada semua pihak yang telah membantu menyumbangkan ide dan pikiran mereka demi terwujudnya makalah ini. Akhirnya saran dan kritik pembaca yang dimaksud untuk mewujudkan kesempurnaan makalah ini penulis sangat hargai.

Badung, 19 November 2020

Penyusun

DAFTAR ISI

KATA PENGANTAR.....	ii
DAFTAR ISI.....	iii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
BAB II GAMBARAN APLIKASI.....	2
2.1 Aplikasi Text Search Berbasis Web.....	2
2.2 Perancangan Mesin NFA.....	2
2.2.1 Pendefinisian Tuple NFA	2
2.2.2 Perancangan Diagram Transisi NFA	5
2.2.3 Proses Pengecekan Dokumen	5
2.3 Fitur Utama Aplikasi.....	6
2.4 Arsitektur Sistem.....	7
2.5 Tampilan Antarmuka	8
BAB III PENGGUNAAN SISTEM.....	10
3.1 Penggunaan Aplikasi Text Search.....	10
BAB IV EKSPERIMEN DAN HASIL	14
4.1 Pengujian Black Box.....	14
4.1.1 Pengujian Fitur Search	14
4.1.2 Pengujian Fitur Machine	14
4.1.3 Pengujian Fitur Evaluation.....	14
4.1.4 Pengujian Fitur Collection	15
4.2 Pengujian Akurasi	15
4.2.1 Pengujian dengan Menambahkan Koleksi Dokumen	15
4.2.2 Pengujian dengan Merubah Koleksi Document.....	16
BAB V PENUTUP.....	17
5.1 Kesimpulan	17

BAB I

PENDAHULUAN

1.1 Latar Belakang

Seiring dengan perkembangan informasi, masalah utama telah bergeser dari cara mengakses informasi menjadi memilih informasi utama yang berguna secara selektif. Pencarian atau pemilihan informasi ini tidak mungkin dilakukan secara manual karena kumpulan informasi yang sangat besar, banyak, dan beragam. Dibutuhkan suatu sistem otomatis untuk membantu user (pengguna) dalam menemukan informasi. *Search engine* (mesin pencari) dapat mengatasi permasalahan tersebut secara efektif. Setiap *search engine* menggunakan proprietary algorithm untuk menciptakan indeks-indeks yang ditampilkan dalam bentuk hasil pencarian (Sarwono, 2010:1).

Sistem pencarian dokumen umumnya menampilkan hasil pencarian berdasarkan kata kunci (*keywords*) dan peringkatnya yang ditampilkan dalam daftar yang panjang. Didalam dokumen teks, tulisan yang terkandung merupakan bahasa dengan struktur yang kompleks dan memuat jumlah kata yang banyak. Mesin pencari menggunakan indeks (yang sudah dibuat dan disusun secara teratur) untuk mencari file setelah pengguna memasukkan kriteria pencarian. *Indexing* atau pengindeksan merupakan proses membangun basis data indeks dari koleksi dokumen. *Indexing* dilakukan terhadap dokumen sebelum pencarian dilakukan.

Berdasarkan konsep *search engine* tersebut dirancanglah suatu aplikasi *text search* berbasis web dimana dalam penerapan untuk *text search* pada aplikasi yang dikembangkan ini tidak menggunakan metode *indexing* melainkan proses pencariannya dengan mengimplementasikan *Non-Deterministic Finite Automata* (NFA). *Non - Deterministic Finite Automata* (NFA) merupakan *Finite Automata* dimana setiap input alphabet/symbol dari suatu state memungkinkan untuk bertransisi ke satu atau lebih dari satu state lain atau bahkan tidak sama sekali.

BAB II

GAMBARAN APLIKASI

2.1 Aplikasi Text Search Berbasis Web

Aplikasi *text search* berbasis web yang dibangun merupakan aplikasi yang bertujuan untuk melakukan pencarian berupa text pada suatu koleksi dokumen dimana proses pencariannya dengan mengimplementasikan *Non-Deterministic Finite Automata (NFA)*. Aplikasi ini dapat menerima inputan berupa keywords dan koleksi dokumen dimana koleksi dokumen ini dapat dihapus dan ditambahkan dengan cara mengupload file-file dokumen dalam format .txt dengan batasan jumlah maksimal file yang bisa diupload yaitu 20 file per satu kali upload. Selanjutnya sistem akan menyimpan file-file tersebut ke dalam suatu direktori/folder.

Proses pencarian yang dilakukan aplikasi *text search* ini, diawali dengan keywords yang diinputkan oleh user baik itu terdiri dari satu kata atau lebih, kemudian berdasarkan *keywords* tersebut sistem akan membangun mesin NFA. Selanjutnya berdasarkan mesin NFA yang telah dibangun tersebut sistem kemudian melakukan pencarian ke setiap file yang terdapat pada direktori/folder yang menampung koleksi dokumen, kemudian sistem akan menampilkan list file yang mengandung paling sedikit 1 *keyword* sebagai hasil dari pencarian.

2.2 Perancangan Mesin NFA

Perancangan mesine NFA dilakukan berdasarkan string *keywords* yang diinputkan oleh user. Misalkan string keywords yang diinputkan user adalah “gubernur koster”, maka langkah-langkah perancangan mesin NFA sebagai berikut :

2.2.1 Pendefinisian Tuple NFA

Mesin NFA didefinisikan dalam 5 tuple yaitu :

- **Q = himpunan state**

Dalam menentukann himpunan state terlebih dahulu menentukan jumlah index himpunan state yang akan digunakan.

Menentukan jumlah index himpunan state dilakukan berdasarkan *keywords* yang diinputkan oleh user.

```
keywords = "gubernur koster"
```

1. Split string masukan keywords berdasarkan spasinya dan masukan ke dalam array.

```
key = [gubernur, koster]
```

2. Kemudian gabungkan setiap kata keyword dalam array menjadi suatu string tanpa spasi

```
keygabung = "gubernurkoster"
```

3. Tentukan jumlah index himpunan state yang digunakan dengan menghitung panjang string gabungan setiap kata keyword sebelumnya

```
index = panjang keygabung = 14
```

4. Kemudian definisikan himpunan state berdasarkan jumlah index yang telah didapat.

```
Q = [q0, q1, q2, q3, q4, q5, q6, q7, q8, q9, q10, q11, q12, q13, q14]
```

- **S = start sate**

Start state selalu didefinisikan untuk q0.

```
S = q0
```

- **F = final state**

Himpunan final state ditetntukan melalui array key dan himpunan state yang telah didapatkan sebelumnya dengan melakukan perulangan.

```
key = [gubernur, koster] → jumlah final state ada 2 berarti terjadi 2 kali perulangan
```

Setiap kata pada array mennetukan state pada himpunan state adalah final state.

```
gubernur → panjangnya 8 → adalah q8
```

```
koster → panjangnya 6 + 8 → adalah q14
```

Sehingga didapatkan :

$$F = [q_8, q_{14}]$$

- **Σ = himpunan simbol input**

Himpunan simbol input dalam NFA yaitu semua karakter ASCII.

$$\Sigma = [\text{semua karakter ASCII}]$$

Setelah menentukan himpunan simbol input NFA selanjutnya tentukan diperlukan juga menentukan simbol input untuk percabangan state. Untuk lebih jelasnya sebagai berikut :

Didalam inputan keywords terdiri dari 2 kata yaitu `key = [gubernur, koster]` setiap kata ini akan membentuk percabangan dari start state yaitu `q0` dan setiap karakter dalam masing-masing kata akan menentukan simbol inputan untuk tiap state dalam percabangan.

➔ `gubernur` : percabangan terdiri dari `q1 – q8`
`q0` |

➔ `koster` : percabangan terdiri dari `q9 - 14`

Sehingga dapat didefinisikan himpunan simbol input untuk percabangan yaitu gabungan dari semua karakter dalam setiap kata kunci dalam bentuk kapital maupun huruf kecil.

$$\Sigma_{\text{percabangan}} = [g, u, b, e, r, n, u, r, k, o, s, t, e, r, G, U, B, E, R, N, U, R, K, O, S, T, E, R]$$

- **δ = fungsi transisi**

Fungsi transisi ditentukan berdasarkan himpunan state dan himpunan simbol input yang telah didapatkan sebelumnya. Fungsi transisi didefinisikan sebagai berikut :

Untuk start state :

- menerima semua himpunan simbol input (karakter ASCII) bertransisi ke dirinya sendiri

$$\delta(q_0, \Sigma) = q_0$$

- menerima inputan `g, G` bertransisi ke `q1`

$$\delta(q_0, g) = q_1$$

$$\delta(q_0, G) = q_1$$

– menerima inputan k,K bertransisi ke q9

$$\delta(q_0, k) = q_9$$

$$\delta(q_0, K) = q_9$$

Untuk percabangan gubernur :

$$\delta(q_1, u) = q_2$$

$$\delta(q_1, U) = q_2$$

----- seterusnya hingga sampai pada q8

$$\delta(q_7, r) = q_8$$

$$\delta(q_7, R) = q_8$$

Untuk percabangan koster :

$$\delta(q_9, o) = q_{10}$$

$$\delta(q_9, O) = q_{10}$$

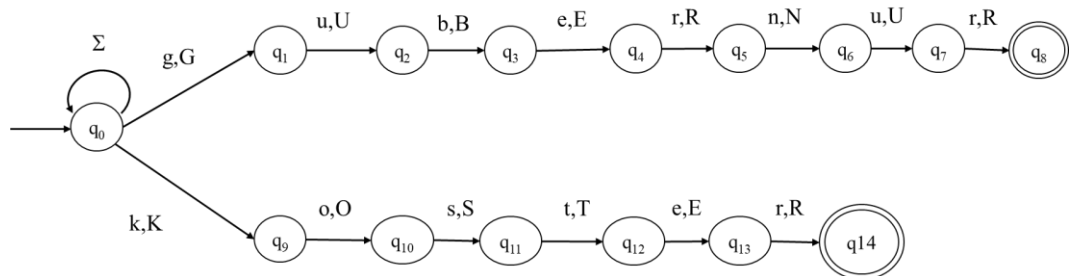
----- seterusnya hingga sampai pada q14

$$\delta(q_{13}, r) = q_{14}$$

$$\delta(q_{13}, R) = q_{14}$$

2.2.2 Perancangan Diagram Transisi NFA

Berdasarkan tupel yang telah didefinisikan seperti di atas dapat dirancang diagram transisi NFA dari inputan keywords = “gubernur koster” sebagai berikut :



2.2.3 Proses Pengecekan Dokumen

Suatu dokumen dalam koleksi dokumen dikatakan *accepted* jika paling sedikit mengandung 1 keyword dari string keywords yang

dimasukan. Proses pengecekan nya dilakukan pada setiap karakter pada dokumen. Proses pengecekan akan berhenti jika pada mesin NFA karakter yang dicek bertransisi ke final state dan karakter tersebut adalah karakter terakhir dalam dokumen atau karakter selanjutnya setelah karakter tersebut adalah spasi atau titik atau koma atau strip atau enter. Untuk lebih jelasnya sebagai berikut :

Keywords = gubernur

xxxxx gubernur xxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxx	xxxxx gubernur. Xxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxx
xxxxx gubernur, Xxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxx	xxxxx gubernur Xxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxx xxxxx gubernur	xxxxx gubernur-bali Xxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxx

2.3 Fitur Utama Aplikasi

Pada aplikasi text search berbasis web yang dibangun memiliki 4 fitur utama yaitu :

1. Fitur Search

- Menerima inputan keywords
- Menampilkan lamanya waktu yang diperlukan dalam melakukan proses pencarian
- Menampilkan list dokumen dari hasil pencarian (nama file + snippet)
- Dapat menampilkan full text dokumen hasil pencarian ketika mengklik nama file tersebut.

2. Fitur Machine

- Menerima inputan keywords
- Menampilkan diagram transisi dari NFA
- Menampilkan Quintuple dari NFA

3. Fitur Evaluation

- Menerima inputan file-file dokumen mana saja yang digunakan (misal dokumen ke 1 – dokumen ke-50).
- Menerima inputan keywords.
- Menampilkan lamanya waktu yang diperlukan dalam melakukan proses pencarian
- Menampilkan list dokumen dari hasil pencarian (nama file + snippet)
- Dapat menampilkan full text dokumen hasil pencarian ketika mengklik nama file tersebut.
- Menerima inputan jumlah dokumen hasil pencarian dan jumlah dokumen yang mengandung paling sedikit 1 keywords
- Menampilkan hasil pengujian akurasi dalam bentuk persen

4. Fitur Doc Collection

- Tempat menambah koleksi dokumen dengan mengupload file-file format .txt dimana maksimal jumlah file yang dapat diupload yaitu 20 file dalam sekali upload.
- Dapat menampilkan seluruh file dalam koleksi dokumen
- Dapat menghapus seluruh file dalam koleksi dokumen

2.4 Arsitektur Sistem

Aplikasi *Text Search* yang dibangun merupakan aplikasi berbasis web. Aplikasi berbasis web merupakan sebuah program atau perangkat lunak yang di akses melalui internet dengan menggunakan web browser. Dalam pembangunan aplikasi *text search* berbasis web ini menggunakan beberapa Bahasa pemrograman yaitu HTML, PHP, CSS.

1. HTML adalah, (HyperText Markup Language) sebuah bahasa standar yang digunakan oleh browser Internet untuk membuat halaman dan dokumen pada sebuah Web yang kemudian dapat diakses dan dibaca layaknya sebuah artikel. HTML juga dapat digunakan sebagai link link

antara file-file dalam situs atau dalam komputer dengan menggunakan localhost, atau link yang menghubungkan antar situs dalam dunia internet.

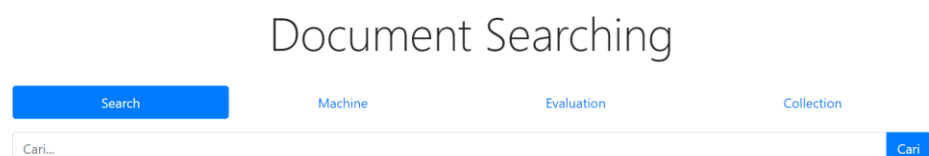
2. Cascading Style Sheet (CSS) merupakan salah satu bahasa pemrograman web untuk mengendalikan beberapa komponen dalam sebuah web sehingga akan lebih terstruktur dan seragam. Pada umumnya CSS dipakai untuk memformat tampilan halaman web yang dibuat dengan bahasa HTML dan XHTML.
3. PHP: Hypertext Preprocessor adalah bahasa skrip yang dapat ditanamkan atau disisipkan ke dalam HTML. PHP banyak dipakai untuk memrogram situs web dinamis. Bahasa pemrograman PHP disini digunakan sebagai back end dari web.
4. JavaScript

JavaScript digunakan pada Web pages untuk meningkatkan design, validate forms, detect browsers, create cookies, GUI dsb. Sama seperti CSS, Javascript tidak dapat berdiri sendiri dan harus didasari oleh HTML atau PHP. Namun perbedaan Javascript dengan CSS adalah, Javascript mengatur logika seperti validasi untuk membuat tampilan website lebih dinamis dan CSS mengatur tampilan dari website tersebut seperti gambar, warna, font, dll.

2.5 Tampilan Antarmuka

Tampilan antarmuka dari aplikasi text search berbasis web adalah sebagai berikut :

1. Fitur Search



2. Fitur Machine

The screenshot shows the 'Document Searching' application with the 'Machine' tab selected. The navigation bar includes 'Search', 'Machine' (active), 'Evaluation', and 'Collection'. Below the navigation bar is a search input field with the placeholder text 'Cari...' and a blue 'Cari' button.

3. Fitur Evaluation

The screenshot shows the 'Document Searching' application with the 'Evaluation' tab selected. The navigation bar includes 'Search', 'Machine', 'Evaluation' (active), and 'Collection'. Below the navigation bar, the section is titled 'Pengujian Akurasi' with a subtitle 'Jumlah koleksi dokumen : 225'. The text 'Masukan dokumen yang akan digunakan :' is followed by two input fields: 'doc ke-' and 'sampai dengan doc ke-'. At the bottom, there is a search input field with the placeholder text 'Cari...' and a blue 'Cari' button.

4. Fitur Collection

The screenshot shows the 'Document Searching' application with the 'Collection' tab selected. The navigation bar includes 'Search', 'Machine', 'Evaluation', and 'Collection' (active). Below the navigation bar, there is a file upload section with a 'Choose Files' button, the text 'No file chosen', and an 'Upload' button. Below this, there are two buttons: 'Lihat Koleksi' and 'Hapus Koleksi'.

BAB III

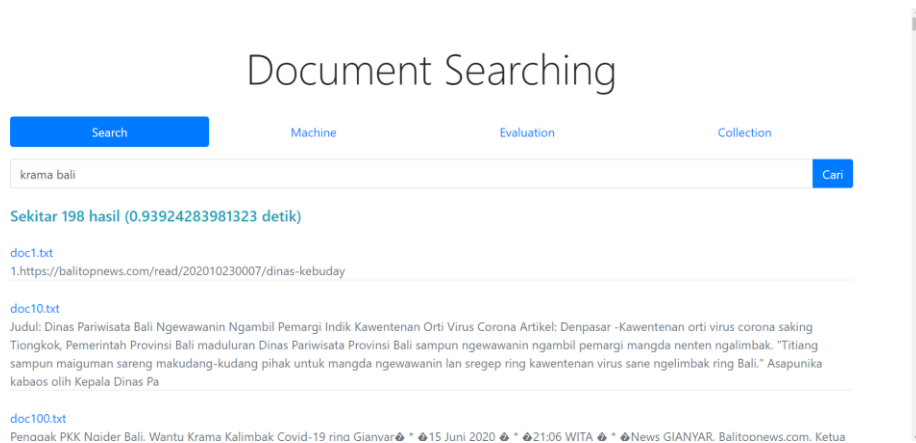
PENGUNAAN SISTEM

3.1 Penggunaan Aplikasi Text Search

Penggunaan dari setiap fitur dalam aplikasi text search berbasis web adalah sebagai berikut :

1. Fitur Search

Pertama-tama ketikan keywords pada *search bar* yang tersedia kemudian klik tombol search maka akan ditampilkan lamanya waktu yang diperlukan dalam melakukan proses pencarian, list dokumen dari hasil pencarian (nama file + snippet), kemudian ketika nama file diklik akan menampilkan full text dari file dokumen.



2. Fitur Machine

Pertama-tama ketikan keywords pada *search bar* yang tersedia kemudian klik tombol search maka akan ditampilkan diagram transisi dan quintuple dari mesin NFA.

Document Searching

Search

Machine

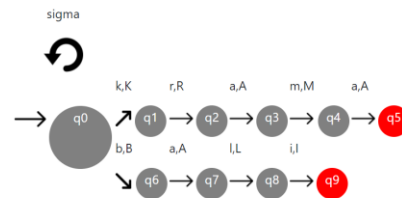
Evaluation

Collection

krama bali

Cari

Diagram Transisi



Quintuple NFA

Q	{ q0,q1,q2,q3,q4,q5,q6,q7,q8,q9. }
Start State	q0
Final State	{ q5,q9. }
Sigma	{ ,_[]{}~!@A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P,Q,R,S,T,U,V,W,X,Y,Z[,],^_~`a,b,c,d,e,f,g,h,i,j,k,l,m,n,o,p,q,r,s,t,u,v,w,x,y,z }.
Delta	q0 --> q0 --> ? q0 --> q0 --> @ q0 --> q0 --> A q0 --> q0 --> B q0 --> q0 --> C q0 --> q0 --> D q0 --> q0 --> E q0 --> q0 --> F q0 --> q0 --> G q0 --> q0 --> H q0 --> q0 --> I q0 --> q0 --> J q0 --> q0 --> K q0 --> q0 --> L q0 --> q0 --> M q0 --> q0 --> N q0 --> q0 --> O q0 --> q0 --> P q0 --> q0 --> Q q0 --> q0 --> R q0 --> q0 --> S q0 --> q0 --> T q0 --> q0 --> I

q0 --> q0 --> h
q0 --> q0 --> i
q0 --> q0 --> j
q0 --> q0 --> k
q0 --> q0 --> l
q0 --> q0 --> m
q0 --> q0 --> n
q0 --> q0 --> o
q0 --> q0 --> p
q0 --> q0 --> q
q0 --> q0 --> r
q0 --> q0 --> s
q0 --> q0 --> t
q0 --> q0 --> u
q0 --> q0 --> v
q0 --> q0 --> w
q0 --> q0 --> x
q0 --> q0 --> y
q0 --> q0 --> z
q0 --> q0 --> {
q0 --> q0 --> |
q0 --> q0 --> }
q0 --> q0 --> ~
q0 --> q0 --> .

```

q0 --> q0 --> 
q0 --> q0 --> 
q0 --> q0 --> 
q0 --> q0 --> 
q0 --> q0 --> 
q0 --> q1 --> k
q1 --> q2 --> r
q2 --> q3 --> a
q3 --> q4 --> m
q4 --> q5 --> a
q0 --> q6 --> b
q6 --> q7 --> a
q7 --> q8 --> l
q8 --> q9 --> i
q0 --> q1 --> K
q1 --> q2 --> R
q2 --> q3 --> A
q3 --> q4 --> M
q4 --> q5 --> A
q0 --> q6 --> B
q6 --> q7 --> A
q7 --> q8 --> L
q8 --> q9 --> I

```

3. Fitur Evaluation

Pertama-tama masukan dokumen mana saja yang akan digunakan untuk pengujian, kemudian ketikkan keywords pada *search bar* yang tersedia kemudian klik tombol search maka akan ditampilkan lamanya waktu yang diperlukan dalam melakukan proses pencarian, list dokumen dari hasil pencarian (nama file + snippet) dan ketika nama file diklik akan menampilkan full text dari file dokumen. Selanjutnya inputkan jumlah dokumen hasil pencarian (diisi otomatis system) dan jumlah dokumen yang mengandung paling sedikit 1 keyword (diisi user). Lalu klik tombol hitung maka akan menampilkan hasil pengujian akurasi dalam bentuk persen.

Document Searching

Search
Machine
Evaluation
Collection

Pengujian Akurasi

Jumlah koleksi dokumen : 225

Masukan dokumen yang akan digunakan :

doc ke-

sampai dengan doc ke-

Cari

Sekitar 46 hasil (0.14008498191833 detik)

doc1.txt

1. <https://balitopnews.com/read/202010230007/dinas-kebuday>

daging Pergub No. 80 warsa 2018 ngeninin indik "Perlindungan dan Penggunaan Bahasa, Aksara, serta Penyelenggaraan Bulan Bahasa Bali", Pemerintah Desa Beraban, Kecamatan

[doc142.txt](#)

KECAMATAN SELEMADEG TIMUR NGAMARGIANG BULAN BAHASA BALI Parikrama Bulan Bahasa Bali Kecamatan Selemadeg Timur kalaksanayang ring rahina Anggara Wuku Kuningan, tanggal 25 Februari ring Wantilan Desa Mambang. Parikrama puniki kabungkah olih dane Bapak Camat Selemadeg Timur sane kaangganin olih Bapak Sekcam, I Wayan Suaba. Ring sambrama wacanannyane,

[doc143.txt](#)

BISNIS ABON PINDANG, METU RING MASA GERING AGUNG Gering agung COVID-19 utawi virus corona majanten banget mawesana kaon majeng kasukertan jagat, punika taler sane karasayang ring Bali. Samian baga ngrasayang pawesana k

Hasil Pengujian Akurasi

Jumlah doc hasil pencarian

46

Jumlah doc mengandung keyword

46

hitung

100.00 %

4. Fitur Collection

Menambah koleksi dokumen dengan mengupload file-file format .txt dimana maksimal jumlah file yang dapat diupload yaitu 20 file dalam sekali upload. Pertama-tama klik choose file, setelah memilih file yang akan diupload lalu klik tombol upload.

Dapat menampilkan seluruh file dalam dalam koleksi dokumen dengan mengeklik tombol lihat koleksi. Dan dapat menghapus seluruh file dalam dalam koleksi dokumen dengan mengeklik tombol hapus koleksi.

Document Searching

Search

Machine

Evaluation

Collection

Choose Files

No file chosen

Upload

Lihat Koleksi

Hapus Koleksi

BAB IV EKSPERIMEN DAN HASIL

4.1 Pengujian Black Box

4.1.1 Pengujian Fitur Search

No.	Skenario pengujian	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1	Mengetikkan keywords pada serach bar kemudian klik tombol search	Sistem akan menerima inputan dan menampilkan lama waktu pencarian, list dokumen dari hasil pencarian (nama file + snippet)	Sesuai harapan	Valid
2	Mengeklik nama file yang berupa link	Sistem akan menampilkan ful text dari dokumen dengan nama file yang yang diklik	Sesuai harapan	Valid

4.1.2 Pengujian Fitur Machine

No.	Skenario pengujian	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1	Mengetikkan keywords pada serach bar kemudian klik tombol search	Sistem akan menerima inputan lalu menampilkan diagram transisi dan quintuple dari NFA	Sesuai harapan	Valid

4.1.3 Pengujian Fitur Evaluation

No.	Skenario pengujian	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1	Masukan dokumen mana saja yang akan diuji. Mengetikkan keywords pada serach bar kemudian klik tombol search.	Sistem akan menerima inputan dan menampilkan lama waktu pencarian, list dokumen dari hasil pencarian (nama file + snippet)	Sesuai harapan	Valid
2	Mengeklik nama file yang berupa link	Sistem akan menampilkan ful text dari dokumen dengan nama file yang yang diklik	Sesuai harapan	Valid

3	meinputkan jumlah dokumen hasil pencarian (diisi otomatis system) dan jumlah dokumen yang mengandung paling sedikit 1 keyword (diisi user). Lalu klik tombol hitung.	Sistem akan menampilkan hasil pengujian akurasi dalam bentuk persen.	Sesuai Harapan	Valid
---	--	--	----------------	-------

4.1.4 Pengujian Fitur Collection

No.	Skenario pengujian	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1	Mengeklik tombol Choose file untuk memilih dokumen yang akan diupload	Sistem akan melakukan browse pada computer untuk memilih dokumen yang akan diupload	Sesuai harapan	Valid
2	Mengeklik tombol upload	Sistem akan menyimpan dokumen ke dalam direktori yang menampung koleksi dokumen setelah mengeklik tombol upload	Sesuai harapan	Valid
3	Mengeklik tombol lihat koleksi	Sistem akan menampilkan isi dari direktori yang menampung koleksi dokumen	Sesuai harapan	Valid
4	Mengeklik tombol hapus koleksi	Sistem akan menghapus semua isi dari direktori yang menampung koleksi dokumen	Sesuai harapan	Valid

4.2 Pengujian Akurasi

4.2.1 Pengujian dengan Menambahkan Koleksi Dokumen

keywords = gubernur koster

No.	Skenario pengujian (penambahan)	Jumlah file yang digunakan	Jumlah list file hasil pencarian (x)	Jumlah file yang mengandung paling sedikit 1 kata kunci (y)	Hasil pengujian ($y/x * 100\%$)
1	doc ke-1–50	50	26	26	100%
2	doc ke-51–100	100	47	47	100%
3	doc ke-101–150	150	66	66	100%
4	doc ke-151–200	200	97	97	100%

5	doc ke-201–225	225	112	112	100%
---	----------------	-----	-----	-----	------

4.2.2 Pengujian dengan Merubah Koleksi Document

keywords = krama bali

No.	Skenario pengujian (pengubahan)	Jumlah file yang digunakan	Jumlah list file hasil pencarian (x)	Jumlah file yang mengandung paling sedikit 1 kata kunci (y)	Hasil pengujian ($y/x * 100\%$)
1	doc ke-1–50	50	46	46	100%
2	doc ke-51–100	50	39	39	100%
3	doc ke-101–150	50	44	44	100%
4	doc ke-150–200	50	46	46	100%
5	doc ke-201–225	25	23	22	96.65%

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil dari pengujian black box yang dilakukan dapat disimpulkan bahwa aplikasi text search berbasis web yang dibangun dengan mengimplementasikan *Non-Deterministic Finite Automata* (NFA) dapat berfungsi dengan benar.

Berdasarkan hasil pengujian akurasi dengan menambah atau mengubah koleksi dokumen yang diuji dapat disimpulkan bahwa pengaruh koleksi dokumen tidak terlalu membawa perubahan pada fungsi dari sistem.

Dari hasil pengujian tersebut dapat diketahui bahwa sistem masih memiliki kekurangan dalam pengecekan keyword dalam dokumen ketika keyword merupakan suffiks dalam suatu string, string tersebut diterima. Misalnya keyword = 'krama', jika terdapat string 'parikrama', maka string tersebut diterima yang seharusnya tidak diterima.

DAFTAR PUSTAKA

Benny Richardson, Kevin Hendy, dkk. (2019). *Penerapan Konsep Non-Deterministic Finite Automata (NFA) pada Aplikasi Simulasi Mesin Kopi Vending*. Jurnal Informatika Universitas Pamulang, ISSN: 2541-1004 Vol. 4, No. 1.

Sarwono, Jonathan. 2010. *SEARCH ENGINE*. Yogyakarta: Penerbit ANDI.

Rizky Indah Melly E.P, W. D. (2012). *Penerapan Konsep Finite State Automata (FSA) pada Mesin Pembuat Minuman Kopi Otomatis*. Jurnal Komputasi, 95-102.