

RANCANG BANGUN APLIKASI PENCARIAN AYAT AL-QUR'AN MENGGUNAKAN METODE ALGORITMA *STRING MATCHING* KNUTH-MORRIS-PRATT BERBASIS ANDROID

Shubhan Ramdhani¹⁾ Anjik Sukmaaji²⁾ Teguh Sutanto³⁾

Program Studi/Jurusan Sistem Informasi
STMIK STIKOM Surabaya

Jl. Raya Kedung Baruk 98 Surabaya, 60298

Email : 1)shubhan.ramdhani@gmail.com, 2)anjik@stikom.edu, 3) teguh@stikom.edu

Abstract: Al-Qur'an is a guide and instructions for Muslims, therefore it is necessary to know the contents of the Al-Qur'an. But there are difficulties in finding a Qur'anic verse that corresponds to the topic that you want to search. One of these factors is there are wide variety of topics contained in a surah and surah have a different topics, while there are 114 surah and 30 juz in 6236 surat in the Qur'an, so if we search manually it will take a lot of time. So we built an application search verses of the Quran by using the Knuth-Morris-Pratt algorithm based on Android. The algorithm used to perform string matching exactly with the order of the characters in the string are matched and highly effective in searching a pattern or patterns of words. This algorithm will perform matching keywords entered by the user with the translation of the Qur'an, and displays the results of our search according to keywords. Based on the test that have been conducted, Knuth-Morris-Pratt algorithms in the process of finding verses of the Qur'an on this application has an average accuracy rate of verse produced is 60% and the rate of speed of searching is 1,25 seconds.

Keywords: ayat al-qur'an, knuth-morris-pratt, pencarian

Seiring berkembangnya teknologi, Al-Qur'an sekarang dapat ditemui dalam bentuk digital, baik berupa teks atau dokumen, maupun dalam bentuk database yang dikemas dalam sebuah aplikasi smartphone, seperti aplikasi Al-Qur'an, tafsir Al-Qur'an, terjemahan Al-Qur'an dan yang lainnya. Al-Qur'an sendiri banyak memuat ilmu pengetahuan yang banyak bermanfaat bagi manusia. Isi kandungan dari Al-Qur'an mengandung hukum-hukum, kisah, akhlak, akidah, dan asas perilaku yang dapat menuntun manusia menjadi lebih baik.

Oleh karena itu sangat perlu untuk mengetahui isi kandungan dari Al-Qur'an, karena merupakan petunjuk bagi umat manusia. Hanya saja terdapat kesulitan dalam mencari ayat Al-Qur'an yang sesuai dengan topik yang ingin dicari. Salah satu faktor tersebut yaitu terdapat berbagai macam topik dalam satu surat dan persamaan topik pada surat yang berbeda. Sedangkan jumlah ayat dan surat pada Al-Qur'an sangatlah banyak, sehingga jika dilakukan pencarian secara manual akan

memakan banyak waktu. Dari banyak aplikasi Al-Qur'an berbasis android yang telah ada, sangat sedikit yang menyediakan fasilitas untuk mencari ayat Al-Qur'an sesuai topik yang ingin dicari. Untuk itu dilakukan penelitian rancang bangun aplikasi pencarian ayat Al-Qur'an menggunakan algoritma string matching Knuth-Morris-Pratt berbasis android.

Algoritma *string matching* merupakan algoritma untuk melakukan pencarian semua kemunculan *string* pendek yang disebut *pattern* pada *string* yang lebih panjang yang disebut teks (Charras, 2004). Beberapa algoritma pencocokan string yang sering digunakan yaitu algoritma Knuth-Morris-Pratt, Brute Force, Rabin Karp dan Boyer-Moore.

Menurut arah pencariannya, algoritma Knuth-Morris-Pratt, algoritma Brute Force, dan algoritma Rabin Karp melakukan pencarian dari kiri ke kanan. Sedangkan algoritma Boyer-Moore melakukan pencarian dari arah yang

berlawanan yaitu kanan ke kiri. Algoritma Rabin Karp dan Boyer-Moore lebih efektif digunakan pada proses pencarian apabila yang dicari adalah kalimat atau lebih dari satu pattern (Wibowo, 2011). Algoritma Knuth-Morris-Pratt sendiri bekerja dengan cara melakukan pergeseran yang lebih sedikit dalam pencocokan string, berbeda dengan algoritma Brute Force yang melakukan pencocokan string dengan pergeseran satu per satu karakter (Okharyadi, 2012). Selain itu algoritma Knuth-Morris-Pratt juga sangat efektif dalam pencarian satu pattern atau pola kata.

Dalam penelitian ini aplikasi yang dikembangkan yaitu aplikasi berbasis android. selain bersifat open source yang memudahkan dalam melakukan pengembangan, juga pengguna smartphone android di Indonesia yang terus meningkat setiap tahunnya. Menurut data yang dilansir International Data Corporation (IDC), untuk Q2 2012, android sendiri menguasai sekitar 52% dari total sistem operasi smartphone yang dipakai di Indonesia, Selain itu harga sebuah smartphone android yang terjangkau. Sebelumnya aplikasi pencarian ayat al-Qur'an sudah ada, namun dalam bentuk aplikasi web. Ada juga dalam bentuk aplikasi android, tetapi menggunakan bahasa Inggris dan hanya menampilkan terjemahan saja. Hal ini tentunya terdapat kendala bagi user yang tidak memahami bahasa Inggris.

Dengan diimplementasikannya algoritma Knuth-Morris-Pratt pada aplikasi pencarian ayat Al-Qur'an ini akan dapat membantu dan mempermudah pengguna dalam proses pencarian ayat-ayat suci Al-Qur'an secara mendalam bagi siapa saja yang membutuhkan.

METODE

Algoritma Knuth-Morris-Pratt

Algoritma Knuth-Morris-Pratt dikembangkan secara terpisah oleh Donald E. Knuth pada tahun 1967 dan James H. Morris bersama Vaughan R. Pratt pada tahun 1966, namun keduanya mempublikasikannya secara bersamaan pada tahun 1977 (Charras, 2004).

Algoritma Knuth-Morris-Pratt (KMP) merupakan algoritma yang digunakan untuk melakukan proses pencocokan string. Algoritma ini merupakan jenis *Exact String Matching*

Algorithm yang merupakan pencocokan string secara tepat dengan susunan karakter dalam string yang dicocokkan memiliki jumlah maupun urutan karakter dalam string yang sama.

Algoritma Knuth-Morris-Pratt menggunakan fungsi pembatas (*border function*) yang digunakan untuk menghitung urutan keberapa perbandingan harus dilakukan. *Border function* dihitung dengan menghitung panjang *prefix* yang ada disebuah pattern yang sama dengan *suffix*-nya (Wibowo, 2011).

Fungsi Pinggiran

Algoritma Knuth-Morris-Pratt melakukan *preprocessing* terhadap *pattern* dengan menghitung fungsi pinggiran. Dengan adanya fungsi pinggiran ini, maka dapat dicegah pergeseran yang tidak berguna, seperti yang terjadi pada algoritma Brute Force.

Fungsi pinggiran $b(j)$ didefinisikan sebagai ukuran awalan terpanjang dari P yang merupakan akhiran dari $P[1..j]$. Sebagai contoh, tinjau pattern $P = \text{agama}$. Nilai F untuk setiap karakter di dalam P adalah sebagai berikut:

j	1	2	3	4	5
$P[j]$	a	g	a	m	a
$b(j)$	-1	0	-1	1	-1

Kompleksitas algoritma pencocokan string dengan Knuth-Morris-Pratt ini dihitung dari kompleksitas untuk menghitung fungsi pinggiran dan pencarian string. Waktu yang diperlukan untuk menghitung fungsi pinggiran yaitu $O(m)$, sementara untuk pencarian string dibutuhkan $O(n)$, sehingga kompleksitas waktu algoritma Knuth-Morris-Pratt adalah $O(m + n)$.

Recall dan Precision

Menurut Hasugian (2006), Perolehan (*recall*) berhubungan dengan kemampuan sistem untuk memanggil dokumen yang *relevan*, sedangkan ketepatan (*precision*) berkaitan dengan kemampuan sistem untuk tidak memanggil dokumen yang tidak *relevan*. Rumus dari *recall* adalah jumlah dokumen relevan yang ditemukan dibagi jumlah semua dokumen yang *relevan*.

$$\text{recall} = \frac{\text{jumlah dokumen retcieve yang relevan}}{\text{total dokumen yang relevan}}$$

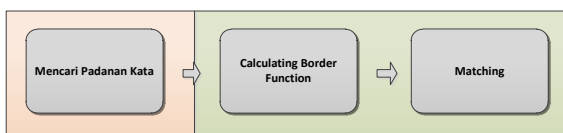
Recall sebenarnya sulit diukur karena jumlah seluruh dokumen yang *relevan* dalam database sangat besar. Oleh karena itu presisi-lah (*precision*) yang biasanya menjadi salah satu ukuran yang digunakan untuk menilai keefektifan suatu sistem temu balik informasi, (Hasugian, 2006).

Precision dapat diartikan sebagai kepersisan atau kecocokan (antara permintaan informasi dengan jawaban terhadap permintaan itu). Jika seseorang mencari informasi di sebuah sistem, dan sistem menawarkan beberapa dokumen, maka kepersisan ini sebenarnya juga adalah relevansi. Artinya, seberapa persis atau cocok dokumen tersebut untuk keperluan pencari informasi, bergantung pada seberapa relevan dokumen tersebut bagi si pencari.

Rumus dari *precision* adalah jumlah dokumen relevan yang ditemukan dibagi jumlah semua dokumen yang ditemukan.

$$precision = \frac{\text{jumlah dokumen retcieve yang relevan}}{\text{total jumlah dokumen yang diretcive}}$$

Pembuatan aplikasi pencarian ayat al-Qur'an menggunakan database sebagai media penyimpanan ayat al-Qur'an dan terjemahnya beserta daftak kata sinonim. Untuk proses pencarian dilakukan dengan cara mencocokkan kata kunci yang diinputkan oleh user dengan terjemahan al-Qur'an yang ada pada database dengan menggunakan algoritma Knuth-Morris-Pratt. Algoritma Knuth-Morris-Pratt memiliki *preprocessing* yaitu *calculating border function* atau menghitung nilai pinggiran dan proses *matching* atau pencocokan. Sedangkan proses mencari padanan kata yaitu mencari kata yang memiliki arti sama dengan kata kunci yang diinputkan oleh user, sebagai contoh kata "shalat", "sholat", dan "solat", ketiganya memiliki arti yang sama. Desain blok diagram untuk gambaran umum aplikasi dapat dilihat pada gambar 2.

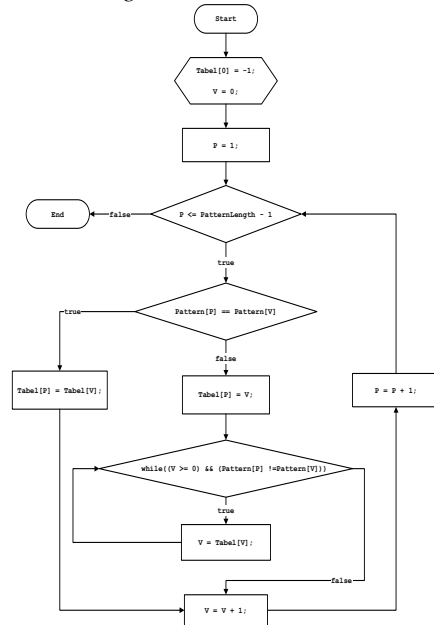


Gambar 1. Blok Diagram Aplikasi Pencairan Ayat Al-Qur'an

a. Mencari Padanan Kata

Mencari padanan kata adalah proses awal pada pencarian ayat al-Qur'an, proses ini bukan merupakan bagian dari algoritma Knuth-Morris-Pratt. Pada proses ini sistem akan mencari daftar padanan kata yang ada pada database kemudian mencocokkannya dengan kata kunci yang diinputkan oleh user. Jika kata kunci yang diinput ada pada daftar padanan kata, maka akan dilakukan pencarian pada semua padanan kata yang sesuai dengan kata kunci. Sebagai contoh, jika user menginputkan kata kunci "sholat" dan daftar padanan kata adalah "sholat", "shalat", dan "solat", maka akan dilakukan pencarian dengan tiga kata padanan tersebut satu per satu. Dan jika kata kunci yang diinput tidak memiliki padanan kata, maka sistem akan langsung melakukan proses selanjutnya, yaitu menghitung nilai pinggiran.

b. Calculating Border Function



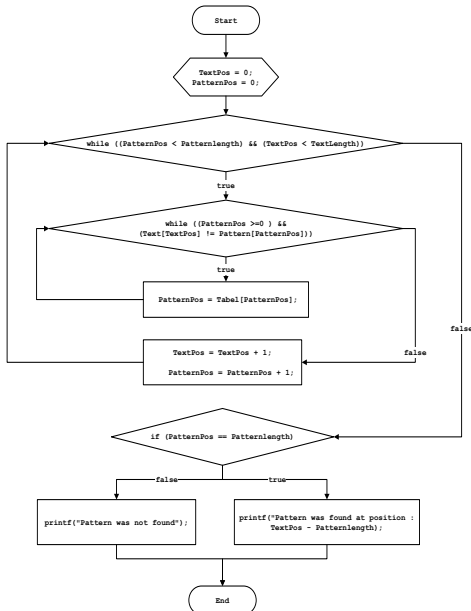
Gambar 2. Flowchart Menghitung Nilai Pinggiran

Tahap kedua dari proses pencarian ayat al-Qur'an yaitu *Calculating Border Function* atau menghitung nilai pinggiran dari pattern atau kata kunci yang telah di inputkan oleh user. Gambar 3 adalah flowchart untuk proses menghitung *border function*.

c. Matching

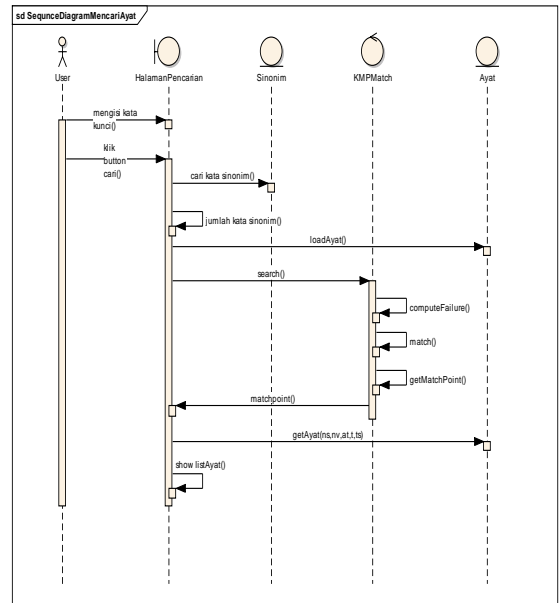
Matching merupakan proses terakhir dari pencarian ayat al-Qur'an. Setelah ditemukan

nilai pinggiran dari pattern tersebut, maka proses selanjutnya yaitu akan dilakukan pencarian string yang diinputkan pada data-data ayat yang ada pada database. Gambar 4. adalah *flowchart* untuk proses *matching*.

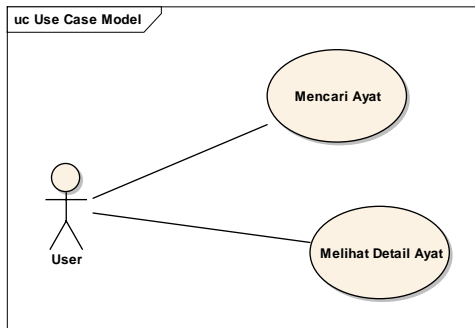


Gambar 3. *Flowchart Matching*

- b. Melihat detail ayat al-Qur'an
- User dapat melihat detail ayat al-Qur'an dari hasil pencarian dari pencarian yang dihasilkan oleh aplikasi beserta terjemahannya.



Gambar 5. *Sequence Diagram Mencari Ayat Al-Qur'an*



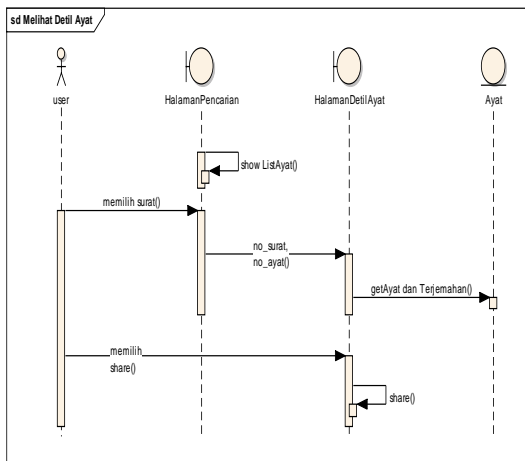
Gambar 4. *Use Case Diagram Aplikasi Pencarian Ayat Al-Qur'an*

Interaksi antara user dengan sistem dapat digambarkan dalam *use case diagram* pada gambar 3.17 Pada gambar tersebut terdapat satu aktor yaitu user sebagai pengguna aplikasi dan dua *use case*, yaitu :

- a. Mencari ayat al-Qur'an

Use case pencarian ayat al-Qur'an adalah proses pencarian dengan menginputkan kata kunci atau ayat yang ingin dicari.

Pada gambar 6 dimulai dari user membuka aplikasi dan menginputkan *keyword* atau kata kunci pada halaman pencarian, kemudian user mengklik *button* cari dan sistem akan mencari padanan kata yang mirip dengan kata kunci, selanjutnya sistem akan mengambil data terjemahan pada database al-Qur'an dan melakukan pencarian. Pencarian sendiri terdiri dari proses menghitung nilai pinggir, dan *matching* atau pencocokan *keyword* dengan *database* terjemahan al-Qur'an, dan mengirimkan *matchpoint* kepada halaman pencarian yaitu pada index berapa terjadi kecocokan tersebut. Kemudian sistem akan mengambil surat beserta ayat yang sesuai dengan hasil pencarian. Kemudian menampilkannya pada halaman pencarian.



Gambar 6. Sequence Diagram Melihat Detail Ayat

Pada Gambar 6 merupakan lanjutan dari sequence diagram mencari ayat Al-Qur'an dimana halaman pencarian sebagai boundary memberikan hasil pencarian kepada user yaitu berupa list surat beserta ayat. Kemudian user memilih salah satu ayat dari hasil pencarian untuk melihat detail dari ayat tersebut, dan sistem akan menampilkan ayat dari entity Al-Qur'an pada halaman detail ayat sesuai dengan nomor surat dan ayat.

HASIL DAN PEMBAHASAN

Kinerja Aplikasi

Uji coba untuk mengukur kinerja aplikasi pencarian ayat al-Qur'an menggunakan algoritma knuth-morris-pratt ini dilakukan dengan dengan 20 kata kunci yang telah ditetapkan. Dari hasil pencarian dengan 20 kata kunci tersebut akan dihitung nilai dari recall, precision, dan tingkat kecepatan aplikasi dalam melakukan pencarian. Tabel 1 merupakan daftar kata kunci yang digunakan pada uji coba ini.

Tabel 1. Daftar Kata Kunci

No	Kata Kunci
1	Qisas
2	Shalat
3	Wasiat
4	Masjidil Haram
5	Puasa
6	Thalaq
7	Iddah
8	Utang Piutang
9	Putus Asa

10	Judi
11	Mahar
12	Riba
13	Nadzar
14	Zabur
15	Kiblat
16	Riya
17	Nikah
18	Sedekah
19	Halal
20	Haram

Hasil uji coba dari pengujian ini dapat dilihat pada Tabel 2. Terdapat 5 point yang dipresentasikan sebagai nilai dari uji coba untuk tiap-tiap kata kunci. Pertama, "ayat diterima" yaitu ayat dari hasil pencarian aplikasi, kedua, "ayat relevan" yaitu ayat yang ada pada index al-Qur'an dan "ayat hilang" yaitu ayat yang ada pada index al-Qur'an tetapi tidak ada pada hasil pencarian aplikasi. Point keempat merupakan nilai precision dari tiap kata kunci. Dan point terakhir yaitu "waktu pencarian" merupakan waktu pencarian pada kata kunci

Tabel 2. Hasil Uji Coba

A	B	C	D	E	F
1	10	7	1	0.60	1.70
2	91	66	15	0.56	2.13
3	10	5	1	0.40	0.95
4	15	9	1	0.53	1.42
5	13	10	0	0.76	0.95
6	26	11	2	0.42	1.69
7	8	8	3	0.62	0.98
8	14	11	0	0.78	1.98
9	18	11	1	0.55	0.99
10	4	3	0	0.75	1.00
11	9	7	2	0.77	1.37
12	19	6	0	0.31	1.11
13	6	5	0	0.83	0.95
14	4	4	0	1.00	0.97
15	5	7	2	1.00	0.99
16	10	4	0	0.40	0.92
17	29	25	11	0.48	1.20
18	12	12	0	1.00	1.51
19	31	7	1	0.19	1.01
20	63	10	1	0.14	1.27
Rata-rata				0.60	1.25

Dimana :

A : Kata Kunci

B : \sum ayat diterima

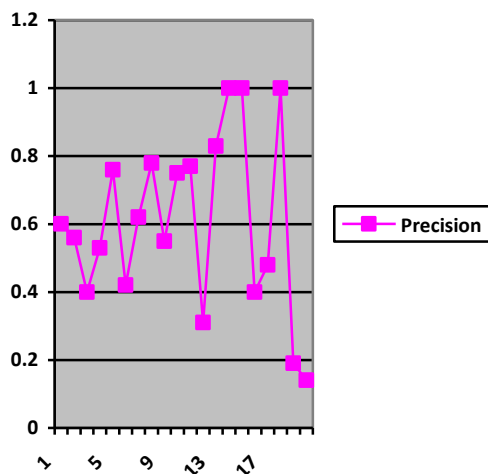
C : \sum ayat relevan

D : \sum ayat hilang

E : Precision

F : Waktu Pencarian(detik)

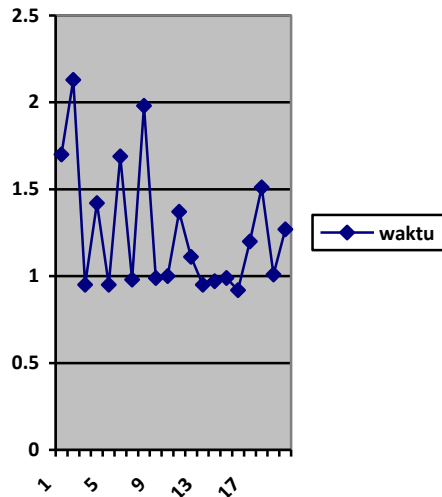
Hasil pengujian nilai *precision* dipresentasikan dengan grafik pada Gambar 7. Dengan sumbu y menyatakan nilai dari precision dan sumbu x menyatakan urutan kata kunci.



Gambar 7. Grafik Pengujian Precision

Dari hasil uji nilai *precision*, hanya terdapat 3 kata kunci yang memiliki nilai *precision* 1, 14 kata kunci dengan nilai *precision* antara 0.4 – 0.8, dan 3 kata kunci dengan *precision* dibawah 0.4, yaitu kata “riba”, “halal”, dan “haram”. Ketiga kata kunci tersebut memiliki nilai *precision* rendah dikarenakan jumlah ayat dari hasil pencarian aplikasi sangat banyak dibandingkan dengan jumlah ayat relevan yang ada pada index al-Qur’an, walaupun ayat yang dihasilkan oleh aplikasi dapat mencakup semua ayat-ayat pada index al-Qur’an. Dari hasil uji coba ini, aplikasi ini hanya memiliki nilai rata-rata *precision* yaitu 0,60 atau sekitar 60%. Hal ini dikarenakan ada beberapa ayat pada kata kunci yang merupakan ayat lanjutan dari ayat sebelumnya. Selain itu juga ada beberapa ayat yang memiliki topik yang sama dengan kata

kunci tapi tidak terdapat kata kunci pada terjemahan ayat tersebut.



Gambar 8. Grafik Grafik Pengujian Kecepatan Algoritma

Pada gambar 8. merupakan grafik hasil pengujian kecepatan pencarian pada 20 kata kunci. Sumbu y menyatakan waktu pencarian dalam detik, dan sumbu x menyatakan urutan kata kunci. Dari hasil uji coba diatas hanya terdapat dua kata kunci yang dengan waktu eksekusi cukup lama yaitu kata “shalat” dan “utang piutang”. Pencarian kata “shalat” membutuhkan waktu 2.13 detik, hal ini dikarenakan terdapat banyak ayat yang mengandung kata “shalat”, selain itu kata “shalat” juga memiliki padanan kata yang cukup banyak sehingga dibutuhkan waktu yang sedikit lama untuk mencari ayat tersebut, begitu juga pada kata kunci “utang piutang” yang memiliki padanan kata yang banyak. Dari hasil uji coba kecepatan pencarian ayat yang dilakukan dengan 20 kata kunci algoritma Knuth-Morris-Pratt memiliki rata-rata pencarian yaitu 1,25 detik.

Dari proses uji coba sebelumnya maka, hasil dari uji coba di rangkum dalam tahap evaluasi ini. Dari uji coba fungsi aplikasi, dapat disimpulkan bahwa aplikasi pencarian ayat Al-Quran menggunakan metode algoritma string matching Knuth-Morris-Pratt ini dapat menghasilkan ayat yang sesuai dengan kata kunci yang diinputkan oleh user sesuai dengan

yang diharapkan berdasarkan uji coba yang telah dilakukan. Tetapi pada aplikasi ini tidak terdapat fitur auto complete dan pembenaran kata karena hanya menangani inputan dari user untuk melakukan pencarian ayat Al-Quran inputan dari user. Selain itu aplikasi ini hanya sebatas menampilkan ayat dan terjemahan Al-Quran, tidak terdapat tafsir dan intisari ayat yang dapat menjelaskan kandungan dari ayat tersebut.

Dari segi kinerja aplikasi, rata-rata ketepatan algoritma Knuth-Morris-Pratt dalam melakukan pencarian ayat sebesar 60% dan rata-rata tingkat kecepatan algoritma Knuth-Morris-Pratt dalam melakukan pencarian ayat adalah 1,25 detik dari uji coba yang telah dilakukan. Untuk itu dapat dilakukan perbandingan dari beberapa algoritma pencarian lainnya untuk mengetahui tingkat efektifitas dari beberapa algoritma pencarian.

KESIMPULAN

Setelah melakukan uji coba dan evaluasi terhadap aplikasi pencarian ayat, dapat ditarik kesimpulan sebagai berikut:

1. Aplikasi pencarian ayat Al-Quran menggunakan metode algoritma string matching Knuth-Morris-Pratt dapat menghasilkan ayat berdasarkan kata kunci yang diinputkan oleh user.
2. Penggunaan algoritma Knuth-Morris-Pratt pada proses pencarian ayat al-Qur'an pada aplikasi ini memiliki tingkat rata-rata ketepatan ayat yang dihasilkan yaitu sekitar 60% dan tingkat kecepatan melakukan pencarian yaitu 1.25 detik.

SARAN

Adapun beberapa saran yang dapat diberikan kepada peneliti berikutnya apabila ingin mengembangkan aplikasi yang telah dibuat ini agar menjadi lebih baik adalah sebagai berikut:

1. Untuk pengembangan selanjutnya aplikasi ini dapat dilengkapi dengan fitur *auto complete* dan pembenaran kata inputan dari *user*.
2. Pengembangan aplikasi ini juga dapat dilengkapi dengan tafsir dan intisari ayat pada tiap ayatnya serta dapat dijalankan pada berbagai macam *platform smartphone*.
3. Diharapkan pada pengembangan selanjutnya dapat membandingkan beberapa algoritma

string matching untuk mengetahui tingkat efektifitas dari beberapa algoritma.

RUJUKAN

- Charras, Christian., & Lecroq , Thierry. (2004). *Handbook of Exact String Matching Algorithm*. London : Colledge Publications.
- Darmawan, Heru. (2010). *Rancang Bangun Aplikasi Search Engine Tafsir Al-Qur'an Menggunakan Teknik Text Mining Dengan Algoritma VSM (Vector Space Model)*. Surabaya: STIKOM
- Hasugian, Jonner. (2006). Penelusuran Ilmiah Secara Online. Pustaka: (Jurnal Studi Perpustakaan dan Informasi) ; Vol.II, No.1. p
- Munir, Rinaldi. (2004). *Strategi Algoritmik*. Program Studi Teknik Informatika, Institut Teknologi Bandung.
- Saputra, Okharyadi (2012). *Makalah IF3051 Strategi Algoritma, Penerapan Algoritma Knuth Morris Pratt dalam Aplikasi Penerjemah Teks*. Program Studi Teknik Informatika, Institut Teknologi Bandung.
- Wibowo, Kevin (2011). *Makalah IF3051 Strategi Algoritma, Perbandingan Algoritma Knuth-Morris-Pratt dan Algoritma Boyer-Moore dalam Pencarian Teks di Bahasa Indonesia dan Inggris*. Program Studi Teknik Informatika, Institut Teknologi Bandung.