

Perancangan Aplikasi Pencarian Undang Undang Otonomi Khusus Aceh (Qanun) Berbasis Android Menggunakan Metode Algoritma Berry Ravindran

Sahminul

Program Studi Teknik Informatika, STMIk Budi Darma, Medan, Indonesia

Email: sahinul134@gmail.com

Abstrak—Qanun adalah seperangkat aturan yang mempunyai kekuatan mengikat dan memaksa umat islam untuk menjalankan aturan-aturan atau hukum. masarakat masih banyak yang belum mengetahui tentang aturan-aturan yang ada didaerah aceh pengertian qanun serta hukuman yang dijatuhkan atas tindakan melanggar qanun, penggunaan buku juga dianggap cukup lama dalam pencarian pasal-pasal pada qanun, sehingga memerlukan proses yang lama dalam menentukan hukuman yang di jatuhkan atas tindakan melanggar qanun. Algoritma Berry Ravindran cara kerjanya terdiri dari dua buah fase, yaitu fase preprocessing dan fase pencarian serta memulai pencocokan pattern dari kiri ke kanan. Algoritma ini sering diimplementasikan dalam berbagai fungsi pencarian string, seperti untuk fungsi Find and Replace, pencocokan DNA, dan lain sebagainya. Untuk fase preprocessing, dilakukan perhitungan untuk menentukan jumlah pergeseran dan setelah itu dilakukan fase pencarian dengan cara melakukan pencocokan string dari arah kiri menuju ke kanan. Apabila terjadi ketidak cocokan, maka pattern akan digeser sesuai dengan aturan algoritma Berry Ravindran. Penulis berharap untuk dapat mengetahui undang-undang khusus aceh qanun berbasis mobile devcie. dan Mempermudah dalam pencarian qanun aceh. Serta agar masyarakat dapat mengetahui hukum di aceh, hanya dengan menggunakan mobile device berbasis android yang sudah mempunyai aplikasi pencarian undang-undang khusus aceh qanun.

Kata Kunci: Qanun, Algoritma Berry Ravindran, Racangan Aplikasi Undang Undang Qanun

Abstract—Qanun is a set of rules that have binding power and force Muslims to carry out the rules or laws. there are still many people who do not know about the existing rules in the area of Aceh, the understanding of qanun and the sentence imposed for violating the qanun, the use of books is also considered to be long enough in the search for articles on the qanun, so it requires a long process of determining the sentence imposed for the actions of violators of the qanun. Berry Ravindran's algorithm works in two phases, namely the preprocessing phase and the search phase and start pattern matching from left to right. This algorithm is often implemented in various string search functions, such as for Find and Replace functions, DNA matching, and so on. For the preprocessing phase, a calculation is performed to determine the number of shifts and after that a search phase is carried out by matching strings from left to right. If there is a mismatch, the pattern will be shifted according to the rules of the Berry Ravindran algorithm. The author hopes to be able to know the special law of Aceh qanun based on mobile devcie. and Facilitate the search for Aceh qanun. And so that the public can know the law in Aceh, only by using an Android-based mobile device that already has a special law search application for Aceh qanun.

Keywords: Qanun, Berry Ravindran Algorithm, Designation of the Law on Qanun

1. PENDAHULUAN

Qanun sendiri dalam Kamus Besar Bahasa Indonesia dikenal dengan nama: Kanun, yang artinya adalah : undang-undang, peraturan, kitab undang-undang, hukum dan kaidah. adapun pengertian Qanun menurut kamus Bahasa Arab adalah: undang-undang, kebiasaan atau adat. Jadi dapat disimpulkan bahwa pengertian dari Qanun adalah : suatu peraturan perundang-undangan atau aturan hukum yang berlaku di suatu daerah (dalam hal ini di Aceh). Di masyarakat Aceh, penyebutan Qanun terhadap suatu aturan hukum atau untuk penamaan suatu adat telah lama dipakai dan telah menjadi bagian dari kultur adat dan budaya Aceh. Aturan-aturan hukum dan juga adat yang dikeluarkan oleh Kerajaan Aceh banyak yang dinamakan dengan Qanun

Qanun adalah Peraturan Perundang-undangan sejenis Peraturan Daerah yang mengatur penyelenggaraan pemerintahan dan kehidupan masyarakat di Provinsi Aceh. Hukum kaku atau tidak fleksibel menimbulkan kompleksitas dan aneka konflik dalam kehidupan sosial, sehingga diperlukan konsepsi hukum yang dapat diterima dan beradaptasi sesuai dengan pola kehidupan bermasyarakat. Salah satu bentuk pemukiran hukum islam adalah Qanun atau peraturan perundangan. Penetrasi hukum islam kedalam peraturan perundangan di indonesia dapat dibedakan menjadi dua model, yaitu penetrasi hukum islam kedalam peraturan perundangan secara substantif dan tidak dinyatakan secara eksplisit sebagai hukum islam dan penetrasi hukum islam ke dalam peraturan perundangan yang secara eksplisit dinyatakan hukum islam. Qanun dibentuk sebagai suatu aturan yang digunakan oleh masyarakat aceh untuk menyelenggarakan ketertiban diaceh. Induk peraturan di Aceh mengacu kepada undan-undang Qanun.

Masalah yang dihadapi yaitu masyarakat masih banyak yang belum mengetahui tentang aturan-aturan yang ada didaerah aceh pengertian Qanun serta hukuman yang dijatuhkan atas tindakan melanggar Qanun, penggunaan buku juga dianggap cukup lama dalam pencarian pasal-pasal pada Qanun, sehingga memerlukan proses yang lama dalam menentukan hukuman yang dijatuhkan atas tindakan melanggar Qanun.

Algoritma Berry-Ravindran merupakan Algoritma yang melakukan pergeseran dengan memperhitungkan pergeseran karakter buruk yang nilai pergeserannya diperoleh dari fase preprocessing, Algoritma Berry Ravindran memulai pencocokan pattern/kata yang dicari dari kiri ke kanan (left to right). Algoritma Berry – Ravindran

memiliki dua fase yaitu fase preprocessing dan fase pencarian Pattern akan bergeser apabila string dan pattern mengalami ketidakcocokan. Pattern akan berhenti bergeser saat pattern ditemukan pada teks atau saat jumlah nilai pergeseran lebih besar dari selisih panjang teks dan panjang pattern.

Penulis menerapkan algoritma berry ravindran berdasarkan penelitian yang dilakukan oleh Sabrina dengan judul “Implementasi Pencocokan String Menggunakan Algoritma Berry Ravindran pada Aplikasi Kamus Bahasa Indonesia–Simalungun Berbasis Android”, menghasilkan bahwa Algoritma Berry-Ravindran dapat diimplementasikan pada aplikasi kamus Bahasa Indonesia – Simalungun berbasis android dan dapat berjalan dengan baik [1] dan Efelin O. Siburian “Implementation and Comparison of Berry-Ravindran and Zhu-Takaoka Exact String Matching Algorithms in Indonesian-Batak Toba Dictionary” Algoritma Berry - Ravindran dan Zhu - Takaoka dapat diimplementasikan pada aplikasi kamus Bahasa Indonesia-Batak Toba dan dapat dijalankan dengan baik [2]. Berdasarkan dari penelitian tersebut, maka penulis menyimpulkan algoritma berry ravindran cocok digunakan dalam perancangan aplikasi pencarian undang-undang pada Qanun karena algoritma Berry-Ravindra lebih cepat dalam hal mencari string terpanjang dan dapat berjalan dengan baik.

2. METODOLOGI PENELITIAN

2.1 String Matching

String Matching adalah proses pencarian semua kemunculan *query* yang selanjutnya disebut *pattern* ke dalam *string* yang lebih panjang atau teks. Kedua *string* terdiri dari sekumpulan karakter yang disebut alfabet yang dilambangkan dengan Σ (sigma) dan mempunyai ukuran σ (tao). *String matching* dibagi menjadi dua, yakni *exact matching* dan *heuristic* atau *statistical matching*[5].

2.2 Algoritma Berry-Ravindran

Sebelum melakukan proses pencarian, algoritma *Berry – Ravindran* memiliki tahapan *preprocessing* untuk menentukan nilai pergeseran. Pada bab kedua, penulis telah memaparkan tahapan untuk membuat tabel nilai pergeseran tersebut. Maka, tabel pergeseran untuk *pattern* ‘aha’ dapat dilihat pada tabel 1 berikut ini.

Tabel 1. Nilai Pergeseran Pattern ‘Aha’

brBc	A	H	*
A	1	1	1
H	2	5	5
*	4	5	5

Pada proses pencarian pertama *pattern* ‘aha’ dicocokkan dengan teks ‘agat’. Apabila jumlah nilai pergeseran lebih besar dari selisih panjang karakter *pattern* dan teks, maka sistem tidak akan melakukan proses pencarian dan akan langsung mengembalikan nilai nol yang menandakan bahwa teks tidak ditemukan. Selisih panjang karakter ‘agat’ dan ‘aha’ adalah satu. Maka proses pencarian hanya memiliki peluang satu kali pergeseran. Pada tabel 2, teks ‘agat’ mengalami penambahan dua karakter ‘00’ yang untuk menghindari terjadinya *Array Index Out Of Bound*. Untuk menentukan nilai pergeseran, algoritma *Berry – Ravindran* mengambil dua karakter di sebelah kanan dari proses pencocokan *string*. Pada kasus teks ‘agat’ hanya memiliki satu karakter di sebelah kanan dari proses pencocokan *string*, yaitu T. Sehingga, dengan adanya penambahan dua karakter ‘0’ setelah huruf T, dapat diperoleh nilai pergeserannya.

Tabel 2. Pencarian Algoritma *Berry – Ravindran* untuk Teks ‘Adat’

A	G	A	T	0	0
A	H	A			
brBc [T][0] = 5					

Pada tabel 1 nilai pergeseran adalah 5. Nilai pergeseran melebihi selisih dari panjang karakter *pattern* dan teks dimana panjang teks adalah 4 dan panjang *pattern* adalah 3. Maka dari itu, sistem akan mengembalikan nilai nol, menandakan bahwa kata ‘aha’ tidak ditemukan di dalam kata ‘agat’ dan mengambil kata berikutnya untuk dicocokkan [1].

2.3 Sejarah Qanun

Aceh adalah provinsi paling barat di Indonesia. Letaknya di ujung utara pulau Sumatera dan Pulau Sabang, sebagai bagian dari provinsi Aceh menjadikannya sebagai titik paling barat di Indonesia. Ibu kota Provinsi Aceh adalah Kota Banda Aceh. Sempat disebut sebagai Aceh Darussalam (1511–1959) dan kemudian berganti menjadi Daerah Istimewa Aceh (1959–2001), Nanggroe Aceh Darussalam (2001–2009) dan saat ini Aceh (2009–sekarang). Aceh berbatasan dengan Samudera Hindia di bagian Barat, berbatasan dengan Teluk Benggala di sebelah utara, Selat Malaka di sisi timur, dan Sumatera Utara di bagian tenggara dan selatan. Aceh dikenal sebagai salah satu titik tempat dimulainya penyebaran Islam di Indonesia dan memainkan peran penting dalam penyebaran Islam di Asia

Tenggara. Pada awal abad ke-17, Kesultanan. Aceh adalah bangsa terkuat dan termaju di kawasan Selat Malaka pada masanya. Hingga saat ini masyarakat Aceh masih memegang teguh syariat Islam, dibuktikan dengan dijalankannya pemerintahan dan tata sosial sesuai syariah Islam, Dasar hukum pelaksanaan syariat islam di Aceh adalah diundangkan UU no 44 tahun 1999 dan UU no 18 tahun 2001. disebutkan bahwa mahkamah syar'iyah akan melaksanakan syariat islam yang di tuangkan ke dalam qanun [6].

3. HASIL DAN PEMBAHASAN

3.1 Analisa Masalah

Dalam melakukan aktifitas pencarian undang undang pada qanun sering ditemukannya masalah. Salah satunya yaitu sulitnya mengingat pasal pada qanun yang ingin dicari dimana pun dan kapan pun. Untuk itu dibutuhkan aplikasi berbasis *mobile* yang mampu mencari pasal-pasal pada undang undang qanun. Aplikasi Pencarian undang undang qanun merupakan aplikasi berbasis *mobile* yang dirancang dan dibuat untuk memudahkan masyarakat dalam melakukan pencarian undang undang qanun dimana pun dan kapan pun. Adapun undang undang qanun yang dicari pada aplikasi pencarian akan menghasilkan berupa pasal-pasal yang di inputkan pada kolom pencarian. Aplikasi yang terdapat dalam *mobile device* sangat memudahkan para pengguna *mobile device* dalam menyelesaikan tugas-tugasnya. Aplikasi Pencarian undang undang qanun yang akan dirancang dalam *mobile device* tersebut akan sangat memudahkan pengguna *mobile device* untuk melakukan pencarian pasal-pasal pada undang undang qanun. Aplikasi Pencarian pasal yang akan dirancang menggunakan android sebagai sistem operasi untuk menjalankan aplikasi tersebut. Android merupakan *platform* terobosan terbaru pada *mobile device* yang sangat digemari para pengguna *mobile device*, hal ini dikarenakan kinerja sistem pada android mudah untuk dioperasikan dengan tampilan yang menarik. Aplikasi Pencarian pasal-pasal berbasis android dapat memudahkan pengguna *mobile device* dalam melakukan pencarian atau *searching* pasal-pasal pada undang undang qanun kapan saja dan dimana saja tanpa perlu menggunakan komputer. Perancangan aplikasi Pencarian undang undang qanun tersebut akan menggunakan *software IDE Eclipse* sebagai editor kode program dan android SDK (*Software Development Kit*) yang diperlukan untuk mulai mengembangkan aplikasi pada *platform* android dengan menggunakan bahasa pemrograman *Java* serta *Android Development Tools* (ADT) sebagai *plugin* yang didesain untuk *IDE Eclipse* yang memberikan kemudahan dalam mengembangkan aplikasi android dengan menggunakan *IDE Eclipse*.

3.2 Penerapan Metode Forward Chaining

1. Fase Preprocessing

Fase *preprocessing* akan menghasilkan tabel berupa nilai pergeseran yang akan digunakan pada saat fase pencarian teks. Pada fase ini, berlaku beberapa kondisi yang berfungsi untuk menentukan nilai pergeseran pada tabel *brBc* (*BerryRavindran bad-character shift*). Nilai pergeseran ditentukan dari berikut.

$$\text{brBc}[a,b] = \min \begin{cases} 1 & \text{if } x[m-1] = a, \\ m-i+1 & \text{if } x[i] x[i+1] = ab, \\ m+1 & \text{if } x[0] = b, \\ m+2 & \text{otherwise} \end{cases}$$

Keterangan

brBc : *Berry-Ravindran bad-character shift*

x : *pattern*

m : panjang *pattern*

a,b : karakter di dalam teks

i : index dari karakter ke-*n* pada teks

Karakter [*a,b*] adalah dua karakter dari teks secara berurutan yang letaknya berada di paling kanan setelah *pattern*. Ada 4 kondisi dimana nilai pergeseran dapat dicari berdasarkan fase *preprocessing* algoritma *Berry-Ravindran*. Nilai pergeseran bernilai 1 apabila karakter *a* adalah [*m-1*]. Nilai pergeseran bernilai *m-i+1* apabila karakter *a* dan *b* berturut-turut adalah [*i*] dan [*i+1*]. Nilai pergeseran bernilai *m+1* apabila karakter *b*=[0], dan yang terakhir, nilai pergeseran bernilai *m+2* untuk karakter *a* dan *b* lainnya.

Berry Ravindran akan diberikan susunan teks PENGAMALANIBADAH dan *pattern* yang akan dicari dari teks tersebut adalah IBADAH. Dari susunan teks dan *pattern* yang diberikan, diketahui bahwa *pattern* adalah IBADAH dengan panjang *pattern* sebanyak 6 karakter.

1. Pada fase *preprocessing*, *pattern* diurutkan sesuai abjad untuk membuat tabel *Berry Ravindran bad-character shift*. *Pattern* tersebut memiliki karakter sebanyak 6 huruf, yaitu A, A, B, D, H, I dan ditambah simbol * sebagai inisialisasi untuk karakter lain selain 6 huruf tersebut. Untuk proses awal, nilai pergeseran bernilai *m+2* untuk karakter [*a,b*] sehingga akan didapatkan nilai pergeseran seperti pada Tabel 1. berikut ini

Tabel 1. Nilai pergeseran untuk karakter (*a,b*) bernilai *m+2*

brBc	A	B	D	H	I	*
A	8	8	8	8	8	8

brBc	A	B	D	H	I	*
B	8	8	8	8	8	8
D	8	8	8	8	8	8
H	8	8	8	8	8	8
I	8	8	8	8	8	8
*	8	8	8	8	8	8

2. Selanjutnya, pada tahap kedua, nilai pada tabel diganti menjadi $m+1$ untuk setiap karakter $b=[0]$. Artinya, karakter b diganti menjadi karakter awal pada *pattern* sehingga menjadi $brBc [a][I]$. Ketika a bernilai 0, maka $brBc$ akan menjadi $brBc [0][I]$. Dengan demikian, nilai pergeserannya akan berubah seperti pada Tabel 3.2. berikut.

Tabel 2. Nilai pergeseran untuk karakter $b = x[0]$ bernilai $m + 1(a)$

brBc	A	B	D	H	I	*
A	8	8	8	8	7	8
B	8	8	8	8	8	8
D	8	8	8	8	8	8
H	8	8	8	8	8	8
I	8	8	8	8	8	8
*	8	8	8	8	8	8

Ketika nilai $a=1$, maka $brBc$ akan berganti menjadi $brBc [1][I]$ yang memiliki nilai pergeseran sama, yaitu $m+1$. Hal tersebut berlaku seterusnya sampai nilai $a=6$ sehingga didapat nilai $brBc$, yaitu $brBc [6][I]$ seperti pada Tabel 3.

Tabel 3. Nilai pergeseran karakter $b=x[0]$ bernilai $m+1 (b)$

brBc	A	B	D	H	I	*
A	8	8	8	8	7	8
B	8	8	8	8	7	8
D	8	8	8	8	7	8
H	8	8	8	8	7	8
I	8	8	8	8	7	8
*	8	8	8	8	7	8

3. Pada tahap ketiga, nilai pergeseran diganti menjadi $m-i$, dimana i adalah perulangan. Perulangan berlangsung ketika nilai $i=0$ sampai $i<m-i$ atau sampai nilai indeks terakhir dari *pattern*. Jika *pattern* memiliki panjang karakter sebanyak 6, maka *indeks* karakter adalah $6 - 1 = 5$. Ketika nilai $i=0$, maka karakter $[i]$ dan $[i+1]$ berturut-turut adalah karakter I dan B sehingga menjadi $brBc [I][B]=5$ dan nilai pergeseran akan berubah menjadi seperti tabel 4.

Tabel 4. Pergeseran karakter $[a,b]$ adalah $x[i]$ dan $x[i+1]$ bernilai $m-i+1 (a)$

brBc	A	B	D	H	I	*
A	8	8	8	8	7	8
B	8	8	8	8	7	8
D	8	8	8	8	7	8
H	8	8	8	8	7	8
I	8	6	8	8	7	8
*	8	8	8	8	7	8

Perulangan terus berlangsung ketika nilai $i=1$, maka *indeks* berada pada posisi $x[1]$ dan $x[2]$ yang diisi karakter B dan A. Selanjutnya, nilai pergeseran akan berubah menjadi seperti pada Tabel 5. berikut ini.

Tabel 5. Nilai pergeseran karakter $[a,b]$ adalah $x[i]$ dan $x[i+1]$ bernilai $m-i+1(b)$

brBc	A	B	D	H	I	*
A	8	8	8	8	7	8
B	5	8	8	8	7	8
D	8	8	8	8	7	8
H	8	8	8	8	7	8
I	8	6	8	8	7	8
*	8	8	8	8	7	8

Perulangan terus berlangsung ketika nilai $i=2$, maka *indeks* berada pada posisi $x[2]$ dan $x[3]$ yang diisi karakter A dan D. Selanjutnya, nilai pergeseran akan berubah menjadi seperti pada Tabel 6. berikut ini.

Tabel 6. Nilai pergeseran karakter [a,b] adalah $x[i]$ dan $x[i+1]$ bernilai $m-i$ (c)

brBc	A	B	D	H	I	*
A	8	8	4	8	7	8
B	5	8	8	8	7	8
D	8	8	8	8	7	8
H	8	8	8	8	7	8
I	8	6	8	8	7	8
*	8	8	8	8	7	8

Perulangan terus berlangsung ketika nilai $i=3$, maka *indeks* berada pada posisi $x[3]$ dan $x[4]$ yang diisi karakter D dan A. Selanjutnya, nilai pergeseran akan berubah menjadi seperti pada Tabel 7. berikut ini.

Tabel 7. Nilai pergeseran karakter[a,b]adalah $x[i]$ dan $x[i+1]$ bernilai $m-i+1$ (d)

brBc	A	B	D	H	I	*
A	8	8	4	8	7	8
B	5	8	8	8	7	8
D	3	8	8	8	7	8
H	8	8	8	8	7	8
I	8	6	8	8	7	8
*	8	8	8	8	7	8

Ketika nilai $i=4$, maka *indeks* berada pada posisi $x[4]$ dan $x[5]$ yang diisi karakter A dan H sehingga nilai pergeserannya akan terlihat seperti pada Tabel 8. berikut. Perulangan akan berhenti ketika nilai i sudah memenuhi kondisi, dimana $i < m-1$.

Tabel 8. Nilai pergeseran karakter[a,b]adalah $x[i]$ dan $x[i+1]$ bernilai $m-i+1$ (e)

brBc	A	B	D	H	I	*
A	8	8	4	2	7	8
B	5	8	8	8	7	8
D	3	8	8	8	7	8
H	8	8	8	8	7	8
I	8	6	8	8	7	8
*	8	8	8	8	7	8

4. Pada tahap akhir, nilai pergeseran berganti menjadi $[m-1]$ untuk karakter a atau karakter terakhir pada *pattern*. Apabila karakter terakhir pada *pattern* adalah H, maka brBc menjadi brBc [H][b] dan pada baris H, nilai yang dihasilkan adalah 1 sebagaimana terlihat pada Tabel 9.

Tabel 9. Nilai pergeseran karakter a bernilai $x[m-1]$

brBc	A	B	D	H	I	*
A	8	8	4	2	7	8
B	5	8	8	8	7	8
D	3	8	8	8	7	8
H	1	1	1	1	1	1
I	8	6	8	8	7	8
*	8	8	8	8	7	8

Setelah semua tahapan pada fase *preprocessing* selesai, maka akan didapat nilai pergeseran seperti yang ditunjukkan pada Tabel 10. berikut ini.

Tabel 10. hasil pergeseran fase *preprocessing* algoritma Berry Ravindran

brBc	A	B	D	H	I	*
A	8	8	4	2	7	8
B	5	8	8	8	7	8
D	3	8	8	8	7	8
H	1	1	1	1	1	1
I	8	6	8	8	7	8
*	8	8	8	8	7	8

2. Fase Pencarian

Setelah fase *preprocessing* selesai, maka akan dilanjutkan ke fase pencarian. Ketika dimasukkan *pattern* IBADAH untuk menemukan teks PENGAMALANIBADAH, maka nilai pergeseran berfungsi untuk menentukan seberapa banyak pergeseran yang akan dilakukan *pattern* untuk menemukan teks yang dicari. Pencarian teks pertama kali akan dilakukan seperti pada Tabel 11. berikut.

Tabel 11. Fase pencarian teks (a)

T	P	E	N	G	A	M	A	L	A	N	I	B	A	D	A	H
P	I	B	A	D	A	H										

Pada Langkah pertama, *pattern* tidak sesuai dengan teks sehingga nilai pergeseran selanjutnya dicari dari dua karakter setelah *pattern* paling kanan, yaitu A dan L, dimana nilai pergeseran untuk karakter [A,L] adalah $brBc [A][L]=8$. Maka, *pattern* akan bergeser sejauh 8 karakter ke kanan seperti pada tabel berikut.

Tabel 12. Fase pencarian teks (b)

T	P	E	N	G	A	M	A	L	A	N	I	B	A	D	A	H
P									I	B	A	D	A	H		

Pada percobaan kedua, *pattern* masih tidak sesuai dengan teks sehingga pencarian terus dilanjutkan. Nilai pergeseran selanjutnya akan diambil dari karakter [A,H] yang nilainya adalah 2 sehingga *pattern* akan bergeser sejauh 2 karakter ke kanan. Pada tahap ini, *brBc* menjadi $brBc [A][H] = 2$ seperti pada Tabel 13. berikut.

Tabel 13. Fase pencarian teks (c)

T	P	E	N	G	A	M	A	L	A	N	I	B	A	D	A	H
P											I	B	A	D	A	H

Pada percobaan terakhir, *pattern* sesuai dengan teks, dimana *pattern* IBADAH sesuai dengan 6 karakter terakhir pada teks PENGAMALANIBADAH. Algoritma *Berry Ravindran* akan terus bergerak ke kanan hingga karakter teks berakhir, dengan nilai $brBc [0][0] = 8$. Namun, karena pada kondisi tersebut *pattern* sudah mencapai ujung teks, maka proses pencarian berakhir.

3.3 Implementasi

Berikut ini hasil dari implementasi program keseluruhan yang telah di rancang penulis adalah Menu utama dapat dikatakan sebagai antar muka (*user interface*) antara *user* dengan program. Menu utama menampilkan pilihan menu yang telah tersedia pada program. Pada menu utama undang-undang qanun tersedia empat pilihan yaitu menu daftar pasal, pencarian, tentang aplikasi dan keluar, dimana menu daftar pasal berisi *link* ke *form* daftar pasal, menu pencarian berisi *link* ke *form* pencarian, menu tentang aplikasi berisi *link* ke *form* tentang aplikasi dan menu keluar untuk keluar dari aplikasi. Gambar untuk tampilan menu utama dapat dilihat pada gambar 1.



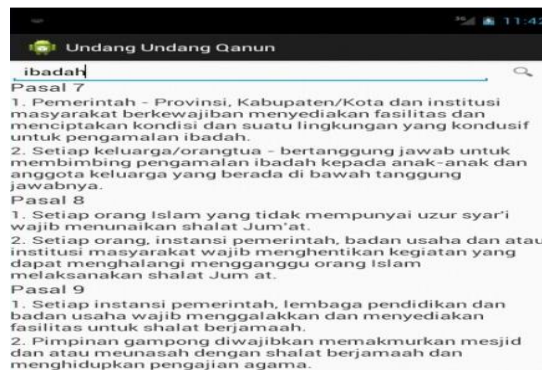
Gambar 1. layout Menu Utama

Form daftar pasal merupakan *form* yang disediakan untuk mencari pasal secara berurutan yang ada dalam undang-undang qanun. *Form* daftar pasal dapat membantu user dalam mencari pasal. *Form* daftar pasal dapat dilihat pada gambar 2.



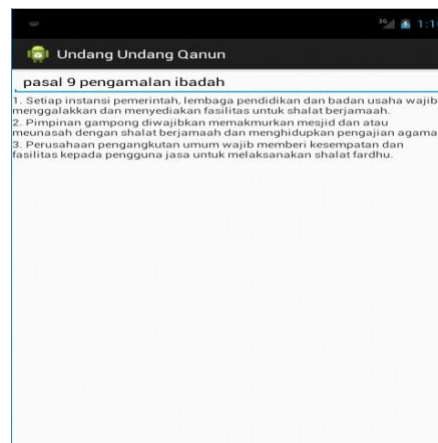
Gambar 2. layout Daftar Pasal

Form Pencarian merupakan *form* yang disediakan untuk mencari pasal yang ada dalam undang-undang qanun tentang ibadah. *Form* pencarian juga dapat membantu user dalam mencari pasal jika user tidak mengingat setiap pasal dari pasal yang di inginkan. *Form* pencarian dapat dilihat pada gambar 3.



Gambar 3. layout Menu Pencarian

Form Hasil Pencarian (isi undang-undang) merupakan *form* yang akan menampilkan hasil pasal yang ada dalam undang-undang qanun tentang ibadah. *Form* hasil pencarian (isi undang-undang) dapat di lihat pada gambar 4.



Gambar 4. layout Menu hasil pencarian (isi undang-undang)

4. KESIMPULAN

Berdasarkan hasil penelitian analisa dan perancangan aplikasi pencarian undang undang qanun maka dapat di tarik beberapa kesimpulan sebagai berikut:

1. Aplikasi pencarian undang undang qanun telah dapat melakukan pencarian Pasal pada aplikasi undang undang qanun.
2. Algoritma *Berry Ravindran* dapat diterapkan dalam perancangan aplikasi undang undang qanun sehingga dapat memudahkan pengguna untuk mencari pasal yang ingin dicari.
3. Aplikasi undang undang qanun telah selesai dirancang dengan menggunakan Software Eclipse Juno sebagai editor, Software Development Kit (SDK) sebagai platform dan telah dapat dijalankan pada perangkat mobile dengan sistem operasi Android 4.3 (*Jelly Bean*).

REFERENCES

- [1] Saut Dohot Siregar, Mawaddah Harahap, and Yohana Marbun, "Aplikasi Penerjemah Kalimat Bahasa Indonesia Ke Bahasa Simalungun Dengan Algoritma Berry - Ravindran," Jurnal Teknovasi, vol. 04, no. 2540-8389, pp. 40-50, 2017.
- [2] Efelin O Siburian, Mohammadandri Budiman, And Jos Timanta Tarigan, "Implementation And Comparison Of Berry - Ravindran And Zhu-Takaoka Exact String Matching Algorithms In Indonesian-Batak Toba Dictionary ," Computing And Applied Informatics (Jocai), Vol. 01, No. 2580-6769, Pp. 39 – 45, 2017.
- [3] suyanto , "atrificial intelegence searching, reasoning, planning, learning, Informatika,Bandung, pp. 3-189, 2014.
- [4] widodo Budiharto Derwin suhartono, atrificial intelligence konsep dan penerapannya. yogyakarta: Andi,pp.3-22,2014,
- [5] Riyanarto Sarno Yeni Anistyasari Rahimi Fitri, "semantic search,"Andi, yogyakarta,pp. 26-189, 2012.
- [6] Rusjdi ali muhammad, "revitalisasi syariat islam di aceh,"ciputat, pp. 4,5-344, 2003.
- [7] A.S, Rosa M. Shalahuddin, "rekayasa perangkat lunak,"Informatika,Bandung,pp. 155-162, 2011.
- [8] zamrony p.juhara, "panduan lengkap pemrograman android,"Andi,Yogyakarta, pp.1-461.2016.

- [9] N. S. H, Pemrograman Aplikasi Mobile Smartphone Dan Tablet Pc Berbasis Android, Revisi Ked. Bandung: Informatika Bandung, 2015.
- [10] Android, "Versi Android," Citra satelit: ©2014 Astrium, DigitalGlobe, 2014. [Online]. Available: <https://www.android.com/>. [Accessed: 24-Jul-2018].
- [11] Fince Tinus Waruwu, Mesran, S.Kom, M.Kom, " Implementasi Algoritma Knuth Morris Pratt Pada Aplikasi Kamus Istilah Latin Flora Dan Fauna Berbasis Android," Informasi dan Teknologi Ilmiah, vol. IV, no. 2, pp. 2339-210X, 2014
- [12] G. P. H. Indra Griha Tofik Isa, "Perancangan Aplikasi Koperasi Simpan Pinjam Berabasis Web (Studi Kasus Koperasi Mitra Setia)," Ilm. Ilmu Ekon., 2017.
- [13] A. A. B. Ginting and D. P. Utomo, "PERANCANGAN APLIKASI CATALOG WISATA DI SUMATERA UTARA MENGGUNAKAN ALGORITMA RABIN-KARP," KOMIK (Konferensi Nasional Teknologi Informasi dan Komputer), vol. 3, no. 1, pp. 57-63, 2019.