

APLIKASI KONVERSI AKSARA LATIN KE AKSARA JAWA MENGUNAKAN FINITE STATE AUTOMATA DENGAN VISUAL BASIC

Candra Sulistya Aji, Eko Adi Sarwoko, Ragil Saputra

Program Studi Teknik Informatika Universitas Diponegoro
Jl. Prof Soedharto, Kampus UNDIP Tembalang Semarang

Email : cand.javu@gmail.com

ABSTRAK

Aksara Jawa merupakan warisan luhur nenek moyang yang perlu dilestarikan terlebih lagi pada era teknologi seperti ini diharapkan Aksara Jawa tetap lestari. Aplikasi Caraka dibuat sebagai sarana untuk memudahkan orang dalam belajar Aksara Jawa dengan lebih mudah dan praktis. Fungsi utama Aplikasi Caraka adalah untuk mengkonversi aksara latin ke Aksara Jawa. Metode *Finite State Automata* dapat menjadi solusi untuk proses konversi aksara latin ke Aksara Jawa. Metode *Finite State Automata* mempunyai fungsi *recognition* sehingga dapat mengenali karakter-karakter aksara latin sebagai masukan. Dalam kasus ini masukan aksara latin bertindak sebagai pita berupa deretan *string* yang nantinya akan dibaca oleh kepala pita (*tape head*). Pola suku kata bertindak sebagai *Finite State Controler* (FSC) yang mengatur jalannya kepala pita untuk proses pemisahan suku kata. Setelah pemisahan suku kata selesai dilakukan, suku kata tersebut akan dikonversikan ke Aksara Jawa sesuai dengan aturan penulisan Aksara Jawa. Fungsi lain dari Aplikasi Caraka adalah menyajikan jenis-jenis Aksara Jawa, menyajikan kumpulan padanan kata Bahasa Jawa, menampilkan tokoh-tokoh pewayangan, dan menyajikan tembang macapat. Bahasa pemrograman yang digunakan adalah *Visual Basic* yang terintegrasi dengan *database Microsoft Access* sebagai media penyimpan data-data mengenai padanan kata, pewayangan dan macapat.

Kata Kunci : Konversi Aksara Jawa, *Finite State Automata*, pemisahan suku kata

1. PENDAHULUAN

Pada era yang memiliki perkembangan teknologi dan informasi sangat pesat seperti sekarang ini, komputer bukanlah sebuah perangkat yang asing. Perkembangan pesat di berbagai bidang tidak pernah terlepas dari peran teknologi khususnya komputer. Pada awalnya, komputer hanya digunakan dalam bidang-bidang khusus, terutama bidang militer yang dimulai pada jaman perang dunia kedua [3]. Komputer dan perangkat lunak dianggap sebagai hal yang rumit. Tetapi seiring perkembangan jaman, komputer selalu memiliki peran penting di setiap bidang kehidupan manusia.

Pada era ini teknologi telah berkembang pesat sehingga ketergantungan manusia akan teknologi semakin besar. Perkembangan teknologi mempunyai banyak manfaat, akan tetapi ada sisi negatif yang akan muncul salah satunya masyarakat akan menerima budaya-budaya asing sehingga budaya mereka sendiri kurang diperhatikan. Aksara Jawa merupakan salah satu warisan luhur budaya Jawa yang perlu dilestarikan.

Salah satu cara untuk melestarikan Aksara Jawa adalah menyediakan media pembelajaran Aksara Jawa secara digital. Dalam Penelitian ini tema yang diambil adalah

tentang konversi dari aksara latin ke Aksara Jawa. Selain itu, skripsi ini juga akan membuat pembelajaran tentang pengenalan Aksara Jawa, *unggah- ungguh basa*, macapat dan pewayangan.

Aksara Jawa bersifat silabik sehingga untuk mengkonversi aksara latin ke Aksara Jawa perlu adanya pemisahan per suku kata (*spelling*) dari aksara latin yang akan dikonversikan. Di dalam ilmu Matematika Diskret terdapat satu cabang ilmu yang khusus mempelajari tentang bahasa, yaitu Teori Bahasa Formal. Bahasa yang dibahas pada Teori Bahasa Formal adalah bahasa tulisan. Keuntungan dari penggunaan Teori Bahasa Formal adalah bisa memodelkan pengenalan suku kata menggunakan *Finite State Automata* [1].

Metode *Finite State Automata* sebelumnya telah digunakan dalam penelitian yang dilakukan oleh Sigit Wasista dan Novita Astin Pada penelitian tersebut, FSA digunakan untuk mengenali karakter dan memisahkan suku kata yang selanjutnya akan dikonversikan ke dalam bentuk suara. Konsep dari penelitian tersebut hampir serupa dengan konsep dari Aplikasi Konversi Aksara Latin Ke Aksara Jawa tetapi ada tiga perbedaan antara kedua konsep tersebut:

1. Masukan konsep penelitian sebelumnya adalah Bahasa Indonesia sedangkan pada penelitian ini berupa Bahasa Jawa.
2. Keluaran dari konsep penelitian sebelumnya berupa suara sedangkan pada penelitian ini adalah teks Aksara Jawa.
3. Diagram transisi konsep yang sebelumnya terdiri dari tiga tingkatan sedangkan pada penelitian ini terdiri dari satu tingkatan.

Dengan menerapkan metode FSA tersebut ke dalam Aplikasi Konversi Aksara Latin Ke Aksara Jawa, masukan teks yang berupa aksara latin akan dikenali dengan menggunakan FSA dan dipisahkan per suku kata sesuai dengan aturan yang telah dibuat. Setelah dipisahkan per suku kata, hasilnya akan dikonversikan ke Aksara Jawa.

2. TINJAUAN PUSTAKA

2.1. *Finite State Automata*

Automata berasal dari bahasa Yunani yaitu *automatos*, yang berarti sesuatu yang bekerja secara otomatis (mesin). Teori *Automata* adalah teori tentang mesin abstrak yang :

1. Bekerja sekuensial
2. Menerima *input*
3. Mengeluarkan *output*

Secara garis besar ada dua fungsi automata dalam hubungannya dengan bahasa, yaitu [5]:

1. Fungsi *automata* sebagai pengenali (*RECOGNIZER*) *string-string* dari suatu bahasa, dalam hal ini bahasa sebagai masukan dari *automata*.
2. Fungsi *automata* sebagai pembangkit (*GENERATOR*) *string-string* dari suatu bahasa, dalam hal ini bahasa sebagai keluaran dari *automata*.

Dalam skripsi ini, pembahasan akan ditekankan pada fungsi pertama dari *automata*. Untuk mengenali *string-string* dari suatu bahasa, akan dimodelkan sebuah *automaton* yang memiliki komponen sebagai berikut [5]:

1. Pita masukan, yang menyimpan string masukan yang akan dikenali.

2. Kepala pita (*tape head*), untuk membaca/menulis ke pita masukan.
3. *Finite State Controller* (FSC), yang berisi status-status dan aturan-aturan yang mengatur langkah yang dilakukan oleh *automaton* berdasarkan status setiap saat dan simbol masukan yang sedang dibaca oleh kepala pita.
4. Pengingat (*memory*), untuk tempat penyimpanan dan pemrosesan sementara.

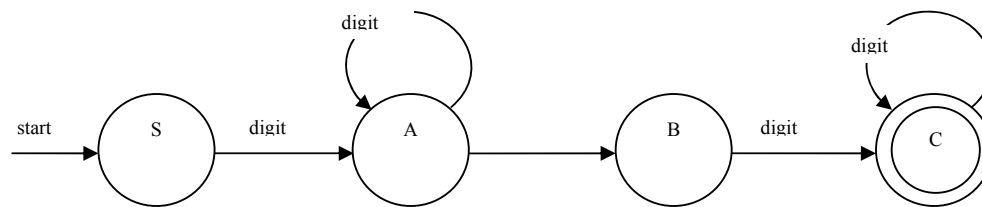
2.2. Finite State Diagram

Finite State Diagram atau diagram transisi, yaitu suatu *graph* berarah yang berfungsi untuk menggambarkan cara kerja suatu FSA[3].

Diagram transisi memiliki komponen sebagai berikut [3]:

1. Setiap simpulnya mewakili setiap status pada FSA.
2. Jika ada transisi dari status p ke status q pada input a , maka ada busur dari p ke q berlabel a .
3. Status awal ditandai dengan kata *START*.
4. Lingkaran bergaris tunggal berarti *state* sementara.
5. *State* akhir ditandai dengan lingkaran bergaris ganda.

Gambar 1 adalah contoh diagram transisi dalam pembacaan bilangan desimal.



Gambar 1. Diagram Transisi

Diagram transisi pada gambar 1 memiliki formal sebagai berikut:

$$M = (Q, V_T, \delta, q_0, F)$$

$$Q = \{S, A, B, C\}$$

$$V_T = \{\text{digit}, .\}$$

$$q_0 = S$$

$$\delta = \text{dapat dilihat pada tabel 1.}$$

$$F = \{C\}$$

Tabel 1. Tabel Transisi FSD

δ	Masukan	
	digit	.
S	A	
A	A	B
B	C	
C	C	

2.3. Struktur Bahasa Jawa

Bahasa Jawa mengenal bahasa tulisan maupun bahasa lisan. Dalam bahasa lisan, dikenal istilah *fonem*, yang merupakan kesatuan bahasa terkecil yang dapat membedakan arti. Fonem dibagi menjadi vokal dan konsonan. Dalam Bahasa Jawa terdapat fonem *ny*, *ng*, *dh*, dan *th*. Sistem fonem Bahasa Jawa tidak mengenal diftong. Kalaupun ada maka kehadirannya merupakan pinjaman dari bahasa lain. Selain itu dalam bahasa Jawa, vokal *e* ada 2 macam yakni *é* dan *è*. Secara garis besar pola suku kata Bahasa Jawa adalah sebagai berikut:

1. V \rightarrow a-bang
2. VK \rightarrow un-tu

- | | | | |
|--------|-----------|----------|------------|
| 3. KV | → ko-bong | 7. KKVK | → nyam-pur |
| 4. KKV | → nya-pu | 8. KVKK | → mang-ga |
| 5. VKK | → ang-gep | 9. KKVKK | → ngang-go |
| 6. KVK | → man-tu | | |

2.4. Font Aksara Jawa

Penulisan Aksara Jawa pada penelitian ini menggunakan *font* Hanacaraka yang dibuat oleh Teguh Budi Sayoga. *Font* Hanacaraka mempunyai kode-kode dari *font* biasa untuk menuliskannya. Perlu adanya penghafalan dari kode-kode untuk bisa menuliskan Aksara Jawa dengan baik. Kode-kode beserta font Aksara Jawa Akan ditampilkan pada tabel 2 sampai dengan 5[9].

Tabel 2. Font Aksara Carakan

Huruf	Pengetikan	Aksara Jawa	Huruf	Pengetikan	Aksara Jawa
Ha	a	ꦲ	Pa	p	ꦥ
Na	n	ꦤ	Dha	d	ꦢ
Ca	c	ꦕ	Ja	j	ꦗ
Ra	r	ꦫ	Ya	y	ꦪ
Ka	k	ꦏ	Nya	v	ꦚ
Da	f	ꦢ	Ma	m	ꦩ
Ta	t	ꦠ	Ga	g	ꦒ
Sa	s	ꦱ	Ba	b	ꦧ
Wa	w	ꦮ	Tha	q	ꦠ
La	l	ꦭ	Nga	z	ꦒ

Tabel 3 Font Aksara Pasangan

Huruf	Pengetikan	Aksara Jawa	Huruf	Pengetikan	Aksara Jawa
Ha	A	ꦲ	Pa	P	ꦥ
Na	N	ꦤ	Dha	D	ꦢ
Ca	C	ꦕ	Ja	J	ꦗ
Ra	R	ꦫ	Ya	Y	ꦪ
Ka	K	ꦏ	Nya	V	ꦚ
Da	F	ꦢ	Ma	M	ꦩ
Ta	T	ꦠ	Ga	G	ꦒ
Sa	S	ꦱ	Ba	B	ꦧ
Wa	W	ꦮ	Tha	Q	ꦠ
La	L	ꦭ	Nga	Z	ꦒ

Tabel 4 Font Aksara Pasangan

Nama Aksara	Aksara Latin	Pengetikan	Aksara Jawa	Nama Aksara	Aksara Latin	Pengetikan	Aksara Jawa
Wulu	i	i	ꦲ	Pengkal	ya	-	ꦲ
Suku	u	u	ꦮ	Cakra	_ra]	ꦲ
Taling	é	[ꦲ	Cakra Keret	_rè	}	ꦲ
Pepet	è	e	ꦲ	Adeg-	Awalan	?	ꦲ

				Adeg	kalimat		
Taling Tarung	o	[o	[o	Panjang Wa	_wa	W	W
Layar	_r	/	/	Panjang La	_la	L	L
Wignyan	_h	h	h	Pada Lungsi	Titik	.	.
Cecek	_ng	=	=	Pada Lingsa	Koma	,	,
Pangkon	Huruf mati	\	\				

Tabel 5 Font Aksara Wilangan

Aksara Latin	Pengetikan	Aksara Jawa	Aksara Latin	Pengetikan	Aksara Jawa
1	1	1	6	6	6
2	2	2	7	7	7
3	3	3	8	8	8
4	4	4	9	9	9
5	5	5	0	0	0

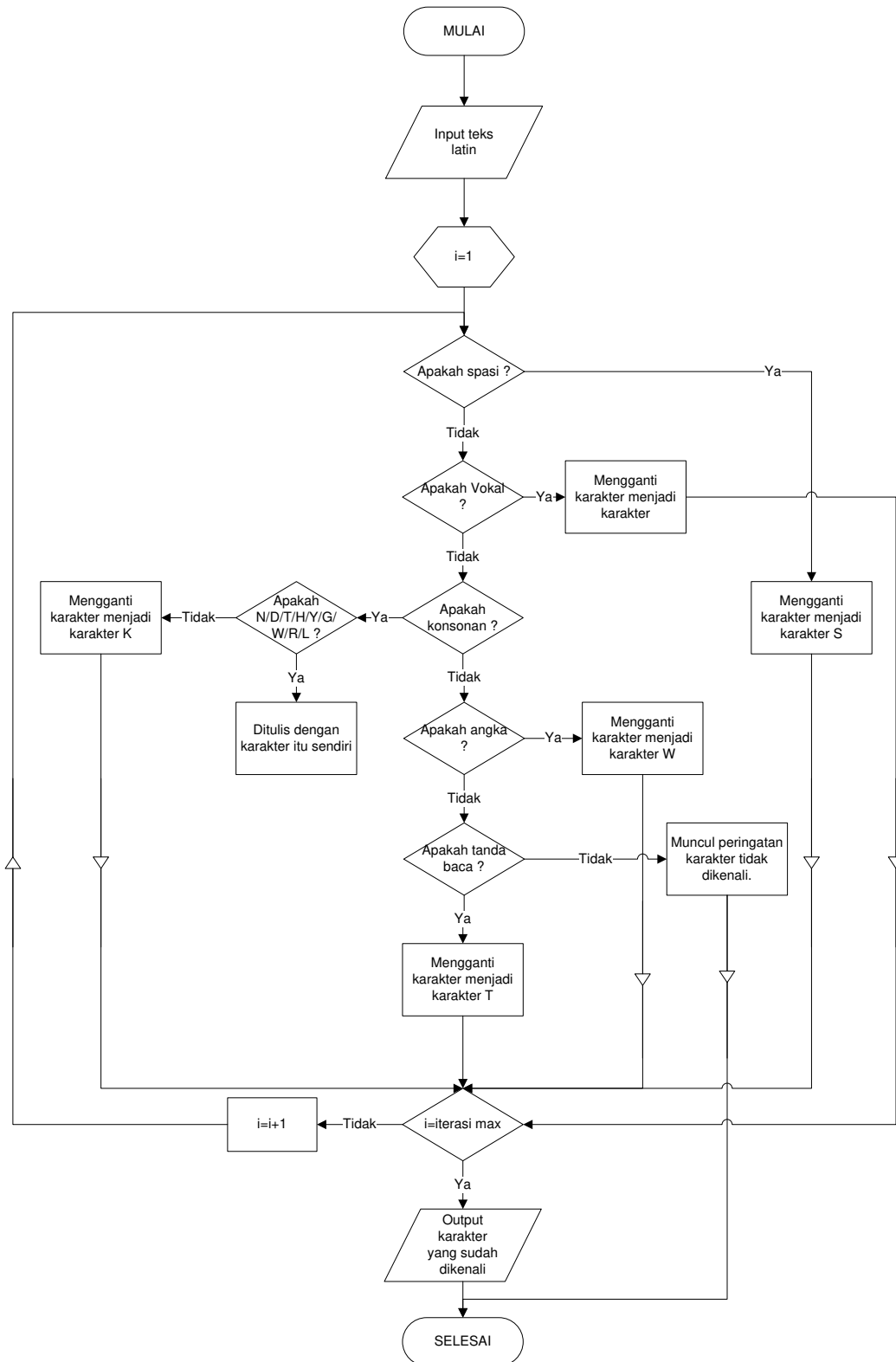
3. PEMBAHASAN

3.1. Analisis dan Desain Sistem

Perancangan aplikasi Caraka merupakan deskripsi dari kebutuhan yang direpresentasikan ke dalam perangkat lunak sebelum dimulai pembuatan *code / coding*. Pada fungsi utama aplikasi Caraka yakni konversi aksara latin ke Aksara Jawa terdapat empat proses utama:

1. Pengenalan karakter

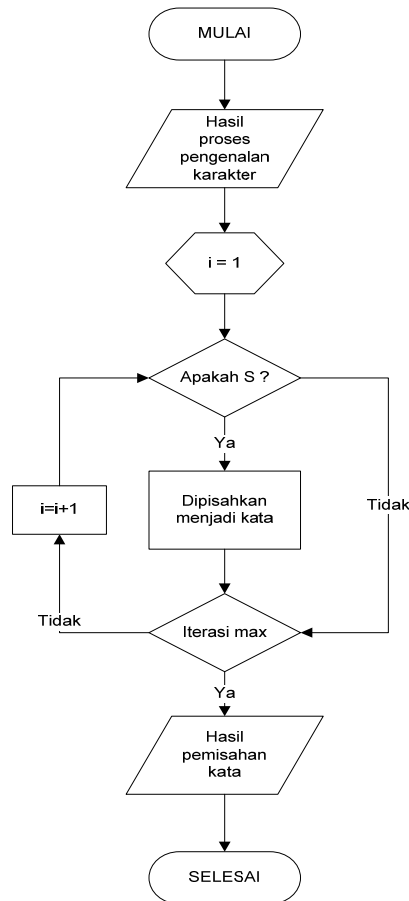
Pada proses pengenalan karakter, teks berupa aksara latin yang dimasukkan oleh *user* akan dikenali jenis karakternya. Pada proses pengenalan karakter, karakter akan dikenali sebagai konsonan, vokal, spasi, tanda baca, atau angka. Khusus untuk konsonan N, D, T, G, Y, W, H, R, dan L akan dikenali sebagai karakter itu sendiri untuk memudahkan proses pemisahan suku kata. Jika ada karakter yang tidak dikenali maka proses akan berhenti dan muncul peringatan karakter tidak dikenali. Proses pengenalan karakter dapat dilihat pada gambar 2.



Gambar 2 Proses Pengenalan Karakter

2. Pemisahan kata

Pada proses pemisahan kata, hasil dari pengenalan karakter akan dipisahkan menjadi beberapa kata. Proses pemisahan kata memanfaatkan pengenalan karakter spasi untuk memisahkan kalimat menjadi beberapa kata. Proses pemisahan kata bisa dilihat pada gambar 3.



Gambar 3 Proses Pemisahan Kata

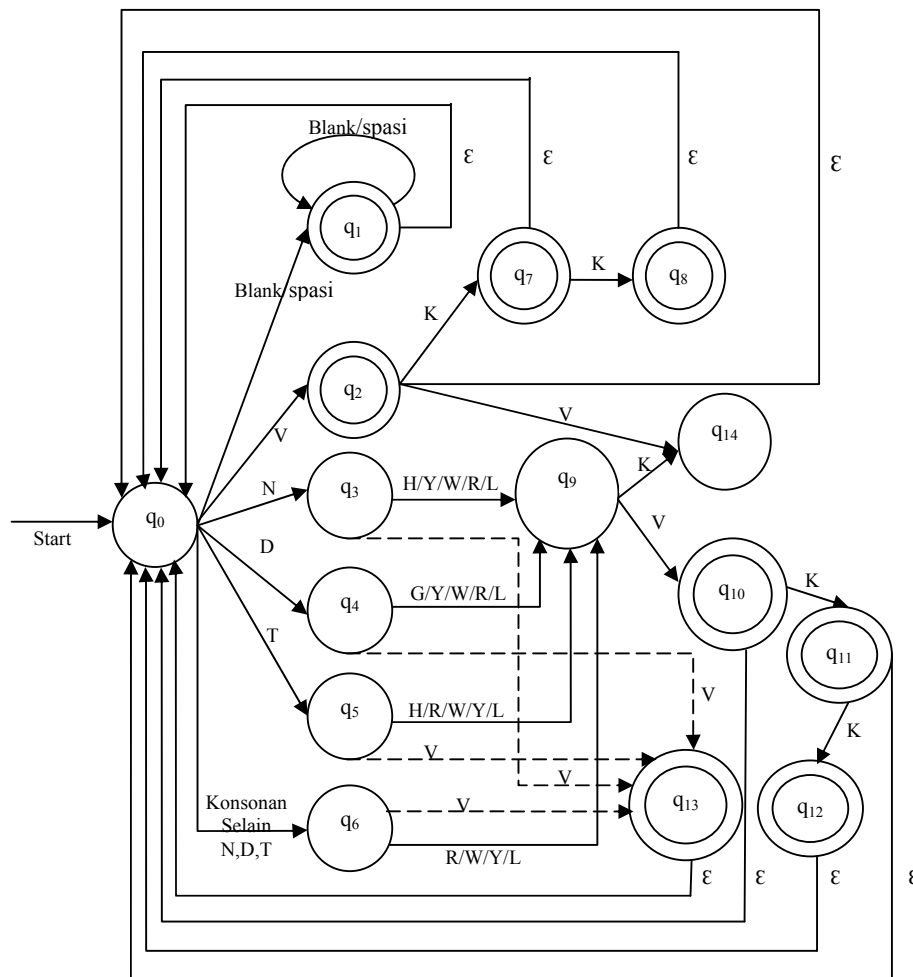
3. Pemisahan suku kata

Pada penelitian yang sudah dilakukan sebelumnya oleh Sigit Wasista dan Novita Astin, proses pemisahan suku kata dibagi menjadi 3 tingkatan diagram transisi. Pada perancangan ini akan dibuat menjadi satu tingkatan diagram transisi. Fungsi dari masing-masing tingkatan pada penelitian sebelumnya adalah sebagai berikut:

1. Tingkatan ke-1 mengenali V, K, dan KV.
2. Tingkatan ke-2 mengenali V, VK, KV, KVK, KKV, KKVK, KKKV dan KKKVK
3. Tingkatan ketiga mengenali VKK, KVKK, dan KKVKK.

Pola suku kata dalam Bahasa Indonesia ada 11 sedangkan pada Bahasa Jawa hanya mengenal 9 pola suku kata sehingga diagram transisi bisa disajikan ke dalam satu tingkatan saja. Penyajian diagram transisi dalam satu tingkatan bisa mempermudah proses

implementasi sehingga waktu yang dibutuhkan untuk menjalankan proses semakin efisien. Diagram transisi pemisahan suku kata pada penelitian ini dapat dilihat pada gambar 4.



Gambar 4 Diagram Transisi Pemisahan Suku Kata

Keterangan :

$$M = (Q, V_T, \delta, q_0, F)$$
$$Q = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8, q_9, q_{10}, q_{11}, q_{12}, q_{13}, q_{14}\}$$

$V_T = \{\text{blank/spasi, V, N, D, T, K(konsonan selain NDT), H, G, Y, R, L}\}$

$$\mathbf{q}_0 = \mathbf{q}_0$$

δ = dapat dilihat pada Tabel 6

$$F = \{ q_1, q_2, q_7, q_8, q_{10}, q_{11}, q_{12}, q_{13} \}$$

Dead state = $\{q_{14}\}$

Tabel 6 Tabel Transisi FSD

δ	Masukan												
	Blank/ Spasi	V	K	N	D	T	Konsonan Selain N/D/T	H	Y	R	W	L	G
q ₀	q ₁	q ₂		q ₃	q ₄	q ₅	q ₆						
q ₁	q ₁												
q ₂		q ₁₄	q ₇										
q ₃		q ₁₃							q ₉	q ₉	q ₉	q ₉	q ₉
q ₄		q ₁₃						q ₉	q ₉	q ₉	q ₉	q ₉	

δ	Masukan												
	Blank/ Spasi	V	K	N	D	T	Konsonan Selain N/D/T	H	Y	R	W	L	G
q ₅		q ₁₃						q ₉	q ₉	q ₉	q ₉	q ₉	
q ₆		q ₁₃							q ₉	q ₉	q ₉	q ₉	
q ₇			q ₈										
q ₉		q ₁₀	q ₁₄										
q ₁₀			q ₁₁										
q ₁₁			q ₁₂										

Pada diagram transisi pengenalan suku kata, pola karakter yang akan dikenali pertama kali adalah blank/spasi, V(vokal), N, D, T, dan K(konsonan selain N, D, dan T). Konsonan N, D, dan T dikenali sebagai karakter itu sendiri karena dalam Bahasa Jawa terdapat fonem TH, DH, NY, NG. Pada diagram transisi ini sudah mencakup 9 pola suku kata: V, VK, KV, KVK, KKV, VKK, KVKK, KKVK, KKVKK. Pada suku kata bahasa indonesia terdapat tiga suku kata tersusun secara berurutan yang terdapat pada kata *pola* (KKKVK-KVK). Pada suku kata Bahasa Jawa tidak mengenal tiga konsonan yang tersusun secara berurutan. Pada kata *ngganggu*(KKKVKK-KV), memang terdapat 3 konsonan yang tersusun secara berurutan (*ngg*) tetapi dalam penulisan Aksara Jawa kata itu akan diubah menjadi *angganggu* (VKK-KVKK-KV) sehingga pola suku kata KKK tidak dikenali dan akan muncul peringatan jika pola tersebut dikenali. Dari penjelasan tersebut maka dalam Bahasa Jawa hanya terdapat dua konsonan yang tersusun secara berurutan. Dua konsonan tersebut terdapat pada fonem TH, DH, NY, dan NG. Selain itu juga ada *sandhangan cakra* yang menyisipkan huruf *r*, *pengkal* yang menyisipkan huruf *y*, *panjang* *la* yang menyisipkan huruf *l*, dan *panjang* *wa* yang menyisipkan huruf *w*, di sela-sela konsonan dan vokal. Pada Bahasa Jawa tidak mengenal diftong sehingga pola suku kata VV tidak dikenali sehingga akan muncul peringatan jika pola tersebut dikenali. Pola-pola ini dirancang sebagaimana ejaan kata dalam Bahasa Jawa.

4. Konversi ke aksara jawa

Proses konversi ke Aksara Jawa adalah proses mengubah aksara latin yang sudah melalui beberapa tahapan proses sebelumnya menjadi Aksara Jawa menggunakan *font* Aksara Jawa. Proses ini akan membaca satu per satu karakter dan akan diubah ke dalam font Aksara Jawa. Ada beberapa aturan dalam proses ini:

1. Apabila ada karakter konsonan ditulis "K\" (karakter K adalah karakter konsonan asli)
2. Apabila ada konsonan diikuti oleh vokal maka:
 - a. Apabila bertemu dengan *a*, maka ditulis "K"
 - b. Apabila bertemu dengan *i*, maka ditulis "Ki"
 - c. Apabila bertemu dengan *u*, maka ditulis "Ku"
 - d. Apabila bertemu dengan *è*, maka ditulis "Ke"
 - e. Apabila bertemu dengan *é*, maka ditulis "[K"
 - f. Apabila bertemu dengan *o*, maka ditulis "[Ko"

3. Ada karakter-karakter konsonan yang mendapat perlakuan khusus:
 - a. Apabila ditemui karakter *ng* maka ditulis “z”
 - b. Apabila ditemui karakter *ny* maka ditulis “q”
 - c. Apabila ditemui karakter *dh* maka ditulis “d”
 - d. Apabila ditemui karakter *th* maka ditulis “v”
4. Apabila karakter konsonan bertemu dengan *r* maka ditulis “KJ”
5. Apabila karakter konsonan bertemu dengan *l* maka ditulis “KL”
6. Apabila karakter konsonan bertemu dengan *y* maka ditulis “K-”
7. Apabila karakter konsonan bertemu dengan *w* maka ditulis “KL”
8. Apabila konsonan bertemu dengan *rè* maka ditulis “K|”
9. Apabila ada pola KV atau V bertemu dengan *r* maka ditulis “KV/” atau “V/” (karakter V mengacu pada aturan 2)
10. Apabila ada pola KV atau V bertemu dengan *h* maka ditulis “KVh” atau “Vh”
11. Apabila ada pola KV atau V bertemu dengan *ng* maka ditulis “KV=” atau “V=”
12. Apabila ada pola K-K (konsonan akhir suku kata pertama diikuti konsonan awal suku kata kedua) maka:
 - a. Konsonan suku kata kedua (termasuk konsonan aturan 3) ditulis dengan huruf kapital.
 - b. Khusus apabila konsonan suku kata pertama adalah *r*, *h*, dan *ng* penulisan konsonan suku kata kedua tetap (tidak huruf kapital)
13. Apabila ada vokal diawal suku kata:
 - a. Apabila vokal *a*, maka ditulis “a”
 - b. Apabila *i*, ditulis “ai”
 - c. Apabila vokal *u*, maka ditulis “au”
 - d. Apabila vokal *è*, maka ditulis “ae”
 - e. Apabila vokal *é*, maka ditulis “[a”
 - f. Apabila Vokal *o*, maka ditulis “[ao”
14. Apabila bertemu angka atau tanda baca ditulis dengan karakter itu sendiri
15. Apabila bertemu dengan _ maka ditulis spasi

Setelah aturan tersebut diterapkan dan selesai, maka akan ditampilkan pada *textbox* dengan format *font* Aksara Jawa sehingga didapatkan tulisan Aksara Jawa dengan pengucapan yang sesuai dengan masukan aksara latin.

3.2. Implementasi

Aplikasi Caraka diimplementasikan dalam bentuk sebuah perangkat lunak berbasis desktop dan dibangun menggunakan bahasa pemrograman *Visual Basic*. Sistem dikembangkan menggunakan *Finite State Automata*.

Tampilan Aplikasi Caraka adalah sebagai berikut:

- | | |
|-----------------------------------|-----------------------|
| 1. Halaman utama | 5. Halaman Pewayangan |
| 2. Halaman Pengenalan Aksara Jawa | 6. Halaman Macapat |
| 3. Halaman Konversi Aksara Jawa | 7. Halaman Bantuan |
| 4. Halaman Padanan Kata | |

Masing-masing halaman bisa dilihat pada gambar 5 sampai 10.

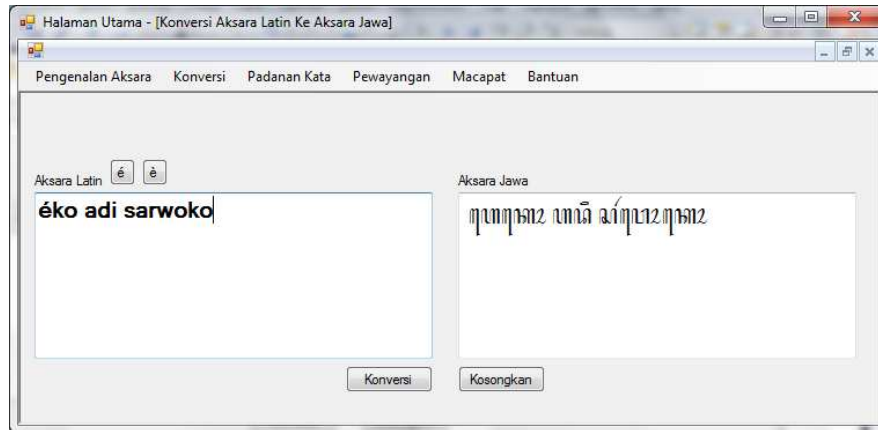


Gambar 5 Form Halaman Utama

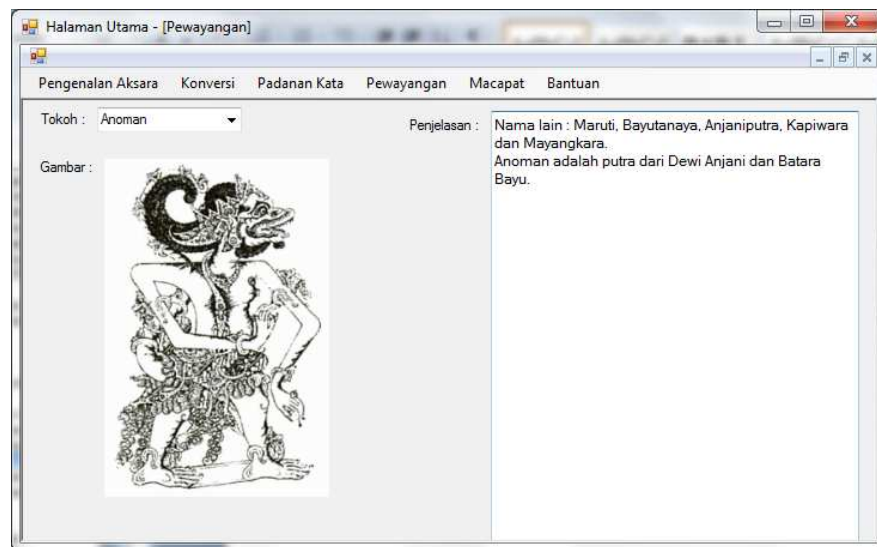
Gambar 6 Form Halaman Pengenalan Aksara Jawa

Nomor	Ngoko	Krama	Krama Inggil
1	Awak	Badan	Salira
2	Anggo	Angge	Agem
3	Mata	Mripat	Tingal, soca, neti
4	Endhas	Sirah	Mustaka
5	Sikil	Suku	Ampeyan
6	Aran	Nama	Asma
7	Pambarep	Pambajeng	Pambajeng
8	Asu	Segawon	Segawon

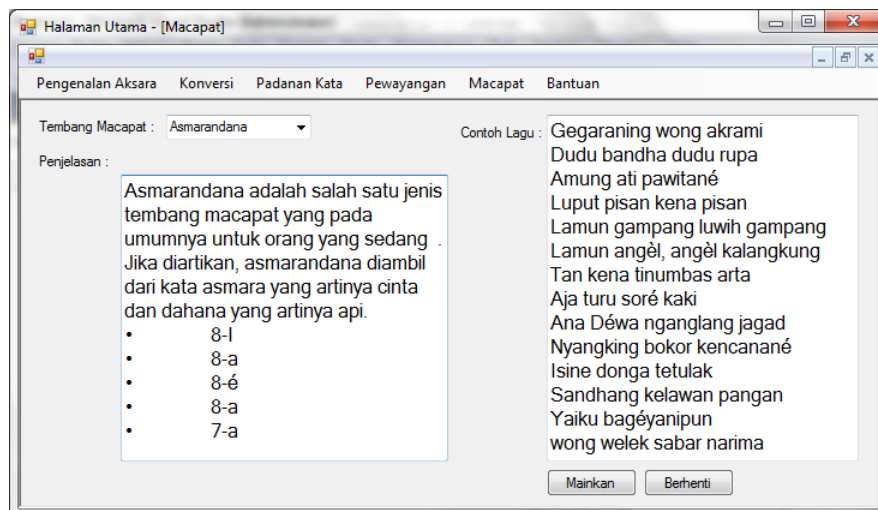
Gambar 7 Form Halaman Padanan Kata



Gambar 8 Form Halaman Konversi Aksara Jawa



Gambar 9 Form Halaman Pewayangan



Gambar 10 Form Halaman Macapat

3.3. Pengujian

Tahap pengujian sistem adalah tahap di mana dilakukan pengujian pada sistem menggunakan berbagai kondisi. Sesuai dengan teori *black-box*, maka sistem akan diuji dari sisi fungsional di mana tiap-tiap fungsi akan dijalankan dengan kondisi tertentu.

Pengujian pada aplikasi ini dilakukan pada sisi fungsional sesuai dengan metode *black-box*. Proses pengujian akan merunut pada *SRS* yang telah dibuat dengan menggunakan butir-butir rencana pengujian. Penjabaran mengenai rencana pengujian dapat dilihat pada Tabel 7.

Tabel 7 Skenario Pengujian Sistem

No	Kelas Uji	Butir Uji	Identifikasi		Tingkat Pengujian	Jenis Pengujian
			SRS	STP		
1.	Pengenal Aksara	Menampilkan Aksara Jawa	SRS-CRK-01	STP-01	Pengujian sistem	<i>Black box</i>
		Mengubah fungsi tombol menjadi Aksara Jawa atau pasangan	SRS-CRK-01	STP-02	Pengujian sistem	<i>Black box</i>
2.	Menerima input teks	Menampilkan masukan teks	SRS-CRK-02	STP-03	Pengujian sistem	<i>Black box</i>
		Memasukkan karakter é dan è	SRS-CRK-02	STP-04	Pengujian sistem	<i>Black box</i>
		Fungsi <i>error</i> huruf e	SRS-CRK-02	STP-05	Pengujian sistem	<i>Black box</i>
		Fungsi <i>error</i> karakter tidak dikenali	SRS-CRK-03	STP-06	Pengujian sistem	<i>Black box</i>
3.	Pengenal karakter	Pengklasifikasian karakter vokal, konsonan, spasi	SRS-CRK-03	STP-07	Pengujian sistem	<i>Black box</i>
4.	Pemisahan kata	Pemisahan karakter per kata	SRS-CRK-04	STP-08	Pengujian sistem	<i>Black box</i>
5.	Pemisahan suku kata	Pemisahan karakter per suku kata	SRS-CRK-05	STP-09	Pengujian sistem	<i>Black box</i>
6.	Konversi ke Aksara Jawa	Mengembalikan karakter asli	SRS-CRK-06	STP-010	Pengujian sistem	<i>Black box</i>
		Menampilkan keluaran Aksara Jawa	SRS-CRK-06	STP-011	Pengujian sistem	<i>Black box</i>
		Mengosongkan <i>textbox</i>	SRS-CRK-06	STP-012	Pengujian sistem	<i>Black box</i>
7.	Padanan Kata	Menampilkan bahasa ngoko, krama madya, dan krama alus pada tabel	SRS-CRK-07	STP-013	Pengujian sistem	<i>Black box</i>
8.	Pewayangan	Memilih tokoh wayang	SRS-CRK-08	STP-014	Pengujian sistem	<i>Black box</i>
		Menampilkan gambar tokoh wayang	SRS-CRK-08	STP-015	Pengujian sistem	<i>Black box</i>
		Menampilkan penjelasan karakter	SRS-CRK-08	STP-016	Pengujian sistem	<i>Black box</i>

No	Kelas Uji	Butir Uji	Identifikasi		Tingkat Pengujian	Jenis Pengujian
			SRS	STP		
		tokoh wayang				
9.	Macapat	Memilih tembang macapat	SRS-CRK-09	STP-017	Pengujian sistem	Black box
		Menampilkan penjelasan tembang macapat	SRS-CRK-09	STP-018	Pengujian sistem	Black box
		Menampilkan contoh bait tembang macapat	SRS-CRK-09	STP-019	Pengujian sistem	Black box
		Memainkan contoh tembang macapat	SRS-CRK-09	STP-020	Pengujian sistem	Black box

4. HASIL DAN PEMBAHASAN

Berdasarkan tabel hasil uji telah dilakukan pengujian untuk seluruh STP dengan mengidentifikasi prosedur pengujian, masukan, dan keluaran yang diharapkan, dan kriteria evaluasi. Hasil proses dari menu utama Aplikasi Caraka yaitu konversi latin ke Aksara Jawa bisa dilihat pada tabel 8.

Tabel 8 Hasil Proses Konversi

No	Teks Asli	Karakter	Kata	Suku Kata	Pengembalian Teks Asli	Aksara Jawa
1.	alu	vk v	vk v	v-kv	a-lu	?alu
2.	untu	vk kv	vk kv	vk-kv	un-tu	?aunT
3.	éko adi sarwoko	vk vsvk vsvk kvkvkv	vk v_v kv_kv kvkvkv	v-kv_v-kv_kv-kv-kv	é-ko_a-fi_sar-wo-ko	?[a[ko a fi s/[wo[ko
4.	ragil saputra	kv kvkskv kvkrv	kv kvkv kvkv kv	kv-kvkv_kv-kv-kv	ra-gil_sa-pu-tra	?rgil\sput]
5.	candra sulistya aji	kv kvkrvsk vk kvkyvsv kv	kv kvkrv kvkv kvkv kvkv kvkv	kv-kv kvkv kvkv kvkv kvkv kvkv	can-fra_su-lis-tya_a-ji	?cnF`sulist- aji

Hasil pengujian pada teks nomor 1-4 telah berhasil diproses secara sempurna. Hasil pengujian pada teks nomor 5 adalah contoh dari proses konversi yang kurang sempurna dikarenakan keterbatasan penulisan font *pengkal* yang seharusnya berada di bawah aksara pasangan *ta*.

5. KESIMPULAN

Dari uraian yang telah dibahas pada bab-bab sebelumnya, maka dapat diambil kesimpulan sebagai berikut :

1. Aplikasi Caraka telah selesai dibangun dan sesuai dengan spesifikasi yang telah direncanakan dengan metode *Finite State Automata*.
2. Proses pengenalan karakter untuk memisahkan suku kata dalam konversi Aksara Jawa dapat dilakukan dengan menggunakan fungsi *recognizer* dari metode *Finite State Automata*.

DAFTAR PUSTAKA

- [1] Basuki, Thomas Anung, 2000, "*Pengenalan Suku Kata Bahasa Indonesia Menggunakan Finite-State Automata*", Jurnal Jurusan Ilmu Komputer, Universitas Khatolik Parahyangan.
- [2] Darusuprpta, dkk, 2003, "*Pedoman Penulisan Aksara Jawa*", Yayasan Pustaka Nusantara, Yogyakarta.
- [3] Hariyanto, Bambang, 2004, "*Teori Bahasa, Automata, dan Komputasi serta Penerapannya*", Informatika Bandung, Bandung.
- [4] Hendra, 2005, "*Dasar Pemrograman Visual Basic*", Indoprog.
- [5] Hopcroft, J.E. dan J.D. Ullman, 1979, "*Introduction to Automata Theory, Languages and Computation*", Addison-Wesley Publishing Company.
- [6] Martono, R.M. Dwijo, 2010, "*Palupi Bahasa Jawa*", Sahabat, Klaten.
- [7] Rusydi, dkk, 1985, "*Kosa Kata Bahasa Jawa*", Depdikbud, Jakarta.
- [8] Rochkyatmo, Amir, 1997, "*Pelestarian dan Modernisasi Aksara Daerah*", Depdikbud, Jakarta.
- [9] Sayoga, Budi Teguh, 2004, "*Dokumentasi dan Panduan Pemakaian Hanacaraka Truetype Font Untuk Perangkat Lunak Komputer Berbasis Sistem Operasi Windows*", Purwokerto.
- [10] Sucipta, Mahendra, 2010, "*Ensiklopedia Tokoh-tokoh Wayang Dan Silsilahnya*", Penerbit Narasi, Yogyakarta.
- [11] Sommerville, I., 2006, "*Software Engineering, Eighth Edition*", Pearson Education Limited, United Kingdom.
- [12] Prasaja, Setya Amrih, "*Kaca Maya Pembelajaran Bahasa Jawa*", diakses dari [http:// http://caraka.jimdo.com/](http://http://caraka.jimdo.com/), tanggal 17 April 2012, pukul 05.55 WIB.
- [13] Pressman, R. S., 2001, "*Software engineering: a practitioner's approach 5th edition*", McGraw-Hill, New York.
- [14] Wasista, Sigit, Novita Astin, 2001, "*Algoritma Sistem Pembaca Teks Bahasa Indonesia Menggunakan Metode FSA (Finite State Automata)*", Jurnal Jurusan Teknik Elektronika Politeknik Elektronika Negeri Surabaya Institut Teknologi Sepuluh Nopember.
- [15] Uma, G. V., 2004, "*Essentials of Software Engineering*", Jaico Publishing House, Mumbai.