

# Rancang Bangun Aplikasi Pembelajaran Baca Tulis Huruf Jawa Dengan Metode Rule Based

Ainatul Mardhiyah, Puji Mahanani, A'la Syauqi

**Abstract**– String Matching translated as sequence of processes that must be passed to resolve the problems in finding a pattern arrangement of character strings with each other exactly in the same structure. Brute-Force Algorithm explain the concept of matching strings in a line theory and source code. Consider the advantages and disadvantages that exist, the expected application of existing theory on Brute-Force algorithm can simplify the process of checking the suitability of the user answers with the answer key for building applications on the Java script learning among students as the students have to pass on the existence of java script. This application need 0.003210462 second for lesson in level 1. And need 0.003294419 second for lesson in level 2, 0.003478261 second for lesson in level 3.

**Keywords:** *String Matching, Brute-Force, Aksara Jawa Learning Application*

**Abstrak** - *String Matching* diterjemahkan sebagai runtutan proses yang harus dilalui untuk menyelesaikan permasalahan dalam menemukan pola susunan karakter *string* yang satu dengan yang lainnya secara tepat dalam struktur yang sama. Algoritma Brute-Force menerangkan tentang konsep pencocokan *string* dalam barisan teori dan *source code*. Mempertimbangkan kelebihan dan kekurangan yang ada, diharapkan penerapan teori yang ada pada algoritma Brute-Force dapat memudahkan proses pengecekan kesesuaian jawaban *user* dengan kunci jawaban untuk membangun aplikasi pembelajaran aksara jawa pada kalangan siswa selaku pelajar yang harus meneruskan eksistensi aksara jawa. Dengan membangkitkan algoritma untuk menghasilkan soal terpilih bagi user, aplikasi ini memerlukan waktu rata-rata sekitar 0.003210462 detik untuk membangkitkan soal terpilih di level 1. Dan dibutuhkan waktu sekitar 0.003294419 detik untuk membangkitkan soal terpilih di level 2. Serta dibutuhkan waktu sekitar 0.003478261 detik untuk membangkitkan soal di level 3.

Manuscript received March 22, 2016. This work was supported in part by Department of Computer Science Universitas Islam Negeri Maulana Malik Ibrahim Malang.

Ainatul Mardhiyah is with the Department of Computer Science Universitas Islam Negeri Maulana Malik Ibrahim Malang, Indonesia. (email [ainadimana@gmail.com](mailto:ainadimana@gmail.com))

Puji Mahanani, was with Department of Computer Science Universitas Islam Negeri Maulana Malik Ibrahim Malang, Indonesia.

A'la Syauqi is the Department of Computer Science Universitas Islam Negeri Maulana Malik Ibrahim Malang, Indonesia. (email [syauqi@ti.uin-malang.ac.id](mailto:syauqi@ti.uin-malang.ac.id))

**Kata kunci:** *String Matching, Brute-Force, Aplikasi Pembelajaran Aksara Jawa*

## I. PENDAHULUAN

Bahasa Jawa memiliki kaidah bagi setiap tata bahasa yang digunakan untuk penggunaannya, diantara berbagai macam kaidah yang harus dipelajari dalam bahasa Jawa, kaidah yang sangat penting dan lumayan sulit untuk dipelajari dan diterapkan dalam keseharian adalah penggunaan Aksara Jawa. Kebiasaan kita mempelajari penulisan huruf balok dan tulisan atau aksara yang digunakan untuk menyatakan aksara bahasa Indonesia menjadikan kita mendapat kesulitan beradaptasi dalam menggunakan aksara Jawa. Dengan mayoritas pengguna bahasa Jawa yang ada, pengguna mayoritas dari baca tulis aksara Jawa sendiri sangat minim jumlahnya. Dengan segala pembiasaan dan pembelajaran yang monoton, pelajar di sekolah pun mulai bosan dengan aksara Jawa. Bahkan, pembelajaran aksara Jawa pernah ditiadakan untuk beberapa waktu dikarenakan kurikulum yang tidak perlu untuk diajarkan dan bahasa Jawa sendiri yang kurang mengglobal.

Menghadapi proses pengikisan pelestarian aksara Jawa, penelitian ini menitik beratkan pada kesulitan pembelajaran aksara Jawa. Dengan menyediakan pemaparan dalam bentuk kalimat, aplikasi ini akan mengajak *user* yang mengoperasikan aplikasi untuk memberikan jawaban input yang sesuai dengan jawaban yang telah dibangkitkan sebelumnya. Sesuai dengan tujuannya untuk menyederhanakan dan menjadikan pembelajaran aksara Jawa menjadi lebih menarik, aplikasi ini akan menyediakan text area yang berisi barisan kalimat yang harus diubah dalam susunan aksara Jawa. Kemudian, aplikasi akan melacak kebenaran jawaban yang diinputkan user dalam text area. Dengan bantuan Algoritma *String Matching* Brute- Force, aplikasi ini akan memunculkan hasil dimana letak kesalahan yang ada pada inputan user. Kemudian user diharuskan menyiapkan pembenaran atau koreksi yang sedemikian rupa jika akhirnya ditemukan kesalahan penulisan pada inputan yang diberikan user dengan kunci jawaban yang ada.

## II. ALGORITMA STRING MATCHING

Algoritma *String Matching* diterjemahkan sebagai runtutan proses yang harus dilalui untuk menyelesaikan permasalahan dalam menemukan pola susunan karakter *string* yang satu dengan yang lainnya secara tepat dalam struktur yang sama. Kamus algoritma dan struktur data, redaksi aslinya berjudul *Dictionary of Algorithms and Data Structures National Institute of Standards and Technology (NIST)*, menyebutkan bahwa pencocokan *string* merupakan langkah yang dilakukan untuk menemukan pola susunan karakter dari permasalahan yang ditemukan. Dalam persyaratannya, karakter yang dicocokkan dengan karakter yang dijadikan acuan harus tepat serupa atau cocok satu dengan lainnya.

*String* adalah susunan dari karakter (alphabet, angka, atau karakter lain) yang biasanya direpresentasikan sebagai struktur data *array*. Sedangkan *string matching* adalah sebuah permasalahan untuk menemukan pola susunan karakter *string* di dalam *string* lain atau bagian dari isi teks. Algoritma pencocokan kata (*string*) atau *String Matching* adalah proses pencarian semua kemunculan *query* yang selanjutnya disebut *pattern* ke dalam *string* yang lebih panjang (teks)[2]. *String matching* dilambangkan dengan rumus :

$$x = x [0...m-1]$$

$$y = y [0...n-1]$$

dimana :  $x = \text{pattern}$   
 $m = \text{panjang pattern}$   
 $y = \text{Teks}$   
 $n = \text{Panjang teks}$

Kedua *string* terdiri dari sekumpulan karakter yang disebut alfabet yang dilambangkan dengan  $\Sigma$  (sigma) dan mempunyai ukuran  $s$  (tau).[1]

Berlandaskan pada kedua definisi di atas, penelitian menggunakan *Exact string matching* berfungsi dengan maksimal bagi proses pencocokan dokumen atau teks yang sama persis dengan *string* masukan dari user. Dengan fungsi tersebut, penelitian ini dianggap lebih maksimal jika penerapan algoritma dalam *Exact String Matching* diimplementasikan sebagai algoritma yang membantu proses pada pembangunan aplikasi pembelajaran aksara Jawa. Proses pencocokan *string* (*string matching*) yang merupakan bagian utama dalam proses pencarian *string* memegang peranan penting untuk mendapatkan teks yang relevan. Dengan demikian hasil dari pencarian sebuah *string* dalam teks tergantung dari teknik atau cara pencocokan *string* yang digunakan.

Algoritma *bruteforce* adalah algoritma yang memecahkan masalah dengan sangat sederhana, langsung, dan dengan cara yang jelas (*obvious way*)<sup>3</sup>. Algoritma Brute-Force menerangkan tentang konsep pencocokan *string* dalam barisan teori dan *source code*. Dalam teorinya, algoritma ini mengklasifikasikan bahwa *text* sebagai kalimat utamaberada pada  $arrayT[1,...,n]$ , kemudian diinisialisasikan lagi variabel lain yaitu *pattern* sebagai kata yang akan dicocokkan pada *text* nantinya berada pada posisi  $arrayP[1,...,m]$ .

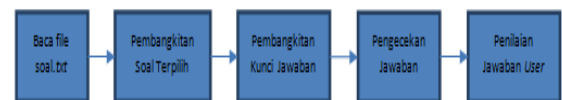
Algoritma *Brute Force* mencocokkan *pattern* (kata yang dicari) dengan semua teks antara 0 dan  $n-m$  untuk menemukan keberadaan *pattern* didalam teks.

## III. DESAIN SISTEM

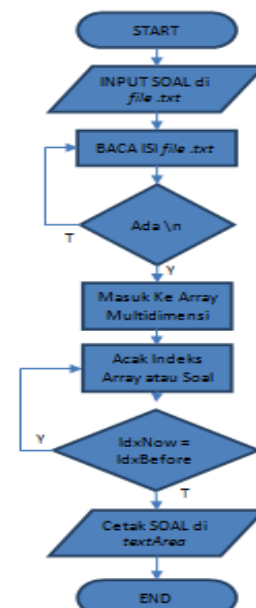
Ilustrasi blok diagram dari aplikasi yang akan dibuat dapat dilihat pada gambar 1.

Gambar 1 menjelaskan serangkaian proses utama yang ada dalam membangun aplikasi pembelajaran aksara Jawa. Dalam ilustrasinya terdapat lima kotak sebagai simbol proses yang menjadi skenario bagi jalannya aplikasi pembelajaran aksara Jawa. Kotak pertamadan kedua menunjukkan, terdapat proses baca bank soal dan pembangkitan daftar soal terpilih dari bank soal, dalam hal ini sistem bertugas melakukan pembangkitan daftar soal terpilih yang akan dijawab oleh *user* dengan sebelumnya melakukan pembacaan dari file soal.txt yang merupakan tempat berkumpulnya soal-soal yang diinputkan oleh penguji atau pun admin dari aplikasi.

Tugas pertama dari suatu sistem adalah menyelesaikan tugas-tugas yang diberikan kepada kelasnya. Dengan peranannya sebagai aplikasi pembelajaran sekaligus sebagai media evaluasi siswa yang menempuh pembelajaran aksara Jawa, media tersebut diharuskan untuk meminimalisir kondisi pengulangan soal yang sama dalam satu kali proses evaluasi atau eksekusi prosesnya. Maka dari itu, diperlukan algoritma yang dapat mencegah pengulangan soal dalam satu kali eksekusi proses yang ada dalam proses operasionalisasi aplikasi pembelajaran aksara Jawa ini.

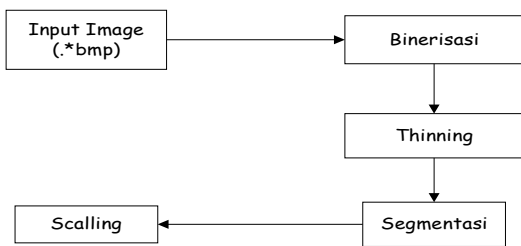


Gambar 1 Desain Blok Diagram Aplikasi Pembelajaran Aksara Jawa

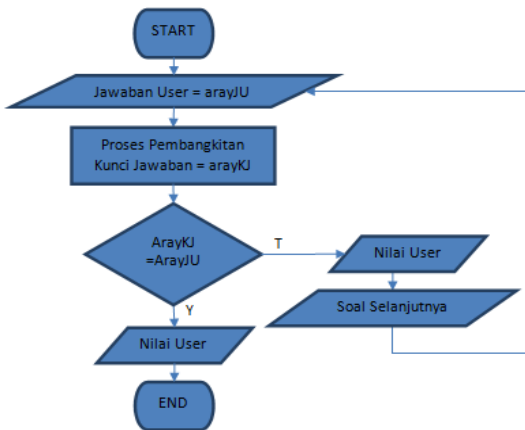


Gambar 2 Flowchart Pembagian Soal Terpilih

Setelah pembangkitan kunci jawaban dieksekusi, aplikasi pembelajaran aksara Jawa akan melakukan pengecekan. Pengecekan ini berhubungan dengan kesesuaian antara kunci jawaban yang dibangkitkan dengan proses transliterasi aksara Jawa dan jawaban yang diberikan user melalui input pada text area. Setelah didapat output berupa benar dan salah, user akan menerima pemberitahuan mengenai letak kesalahan yang dilakukan. Ketika user berhasil memberikan jawaban dengan benar, maka user akan menerima penambahan nilai, namun jika user menerima pesan yang berisi daftar kesalahan inputan jawaban dari user, maka user tidak bisa menerima nilai tambahan. Namun user dapat terus melanjutkan jalannya aplikasi dengan memperbaiki kesalahan inputan yang tidak sesuai dengan kunci jawaban yang telah dibangkitkan. Untuk penggambaran dari proses pada tahap penambahan nilai dari user dijelaskan sesuai pada Gambar 4 mengenai diagram proses penilaian jawaban user.



Gambar 3 Diagram preprocessing



Gambar 4 Diagram Proses Penilaian Jawaban User



Gambar 5 Interface Halaman Depan

IV.HASIL UJICOBA

Sebelum eksekusi proses pertama, user akan disuguhkan halaman depan yang akan menghubungkan user dengan halaman interface kelas lainnya seperti terdapat pada Gambar 5 mengenai interface halaman depan yang berisi 4 menu utama yaitu;

- a. Mulai dolanan, menu ini berisi level-level yang akan dimainkan. Level sederhana, menengah, dan lanjutan
- b. Input prakara, halaman yang dapat digunakan user untuk melakukan input soal sesuai dengan kehendaknya.
- c. Pitulungan, halaman ini berisikan petunjuk untuk menjalankan aplikasi. Dalam halaman ini juga berisi penjelasan mengenai fitur-fitur khusus yang ada dalam aplikasi, misal bagaimana menggunakan keyboard aksara jawa, juga bagaimana menggunakan fungsi input karakter khusus dalam halaman input prakara.
- d. Tutup software, menutup aplikasi yang sedang bekerja

V. KESIMPULAN DAN SARAN

A. Kesimpulan.

Aplikasi pembelajaran aksara jawa yang dibangun menggunakan bahasa pemrograman Java adalah aplikasi pembelajaran yang memiliki tujuan untuk mengajak generasi muda untuk melatih kemampuan tulis menulis aksara jawa. Dengan membangkitkan algoritma untuk menghasilkan soal terpilih bagi user, aplikasi ini memerlukan waktu rata-rata sekitar 0.003210462 detik untuk membangkitkan soal terpilih di level 1. Dan dibutuhkan waktu sekitar 0.003294419 detik untuk membangkitkan soal terpilih di level 2. Serta dibutuhkan waktu sekitar 0.003478261 detik untuk membangkitkan soal di level 3.

Dalam membangkitkan kunci jawaban yang dijadikan sebagai acuan untuk melakukan pengecekan kesesuaian jawaban user, aplikasi ini membangkitkan algoritma decision tree yang dimodifikasi agar menghasilkan susunan aksara jawa sesuai dengan tuntunan aksara jawa lengkap. Dengan penambahan kondisi yang diterapkan bagi karakter vokal a, o dan â yang belum pernah ada pada aplikasi sebelumnya, pembangkitan kunci jawaban pada aplikasi ini dinilai makin kaya dan memiliki tingkat akurasi 100%, maka pembangkitan kunci jawaban dianggap sesuai dengan tujuan dan permasalahan yang diajukan. Dan akurasi dari kunci jawaban yang dibangkitkan tidak meragukan bagi proses selanjutnya.

Setelah kunci jawaban dibangkitkan, dan user selesai memberikan inputan jawaban, maka sistem akan menjalankan pengecekan kesesuaian apakah jawaban user dikatakan benar atau salah. Pada pembahasan sebelumnya, ditemukan bahwa kemampuan sistem menemukan ketidak sesuaian dengan jawaban user. Dari hasil yang diperoleh, prosentase ketepatan

pemberian koreksi dihitung sebesar 100% dari hasil pengujian yang dilakukan.

#### B. Saran.

Membuat produk yang dapat memajukan kesejahteraan dan kehormatan bangsa merupakan upaya utama dari pembangunan aplikasi ini. Dengan mempertimbangkan berbagai nilai yang ada, penulis memiliki beberapa pesan dan saran yang akan memajukan penelitian selanjutnya

1. *Interface* dari keyboard *on-screen* untuk memberikan input harus didesain lebih cantik lagi dan lebih mudah digunakan dari *interface* yang sebelumnya.
2. Tambahkan fasilitas pembagian level yang lebih spesifik untuk semakin menantang kemampuan *user* dalam menjawab soal yang diberikan dan mengasah kelincahan beraksara Jawa.
3. Melihat kemajuan teknologi yang ada, akan sangat disayangkan jika aplikasi ini tidak dikembangkan menjadi aplikasi *mobile* yang berbasis *operating system* android, windows phone, Symbian, maupun iOS.

#### VI.DAFTAR PUSTAKA

- [1] Mulyani. 2014. Implementasi String Matching pada Aplikasi Tafsir Al Qur'an Menggunakan Algoritma Brute Force Berbasis Android. Medan: Program Studi Teknik Informatika STIMIK Budi Darma Medan.
- [2] Riyanto, dkk. 2012. ***Semantic Search Pencarian Berdasarkan Konten.*** Yogyakarta: Andi.
- [3] Munir, R dsn Syaroni M.2005. Pencocokan String Berdasarkan Ucapan (Phonetic String Matching) Dalam Bahasa Inggris. Yogyakarta: Seminar Nasional Aplikasi Teknologi Infprmasi (SNATI).