

APLIKASI SARAN BUKU BACAAN BAGI PENGUNJUNG PERPUSTAKAAN AMIK STIEKOM SUMATERA UTARA BERDASARKAN ALGORITMA *BRUTE FORCE*

Taronisokhi Zebua¹, Natalia Silalahi²

^{1,2}Program Studi Teknik Informatika AMIK STIEKOM Sumatera Utara
^{1,2}Jl. H. Adam Malik No. 18 Rantauparapat, Labuhan Batu, Sumatera Utara
¹taronizeb@gmail.com,²natalia.novena.silalahi@gmail.com

Abstract

Service of information about books in a library is very helpful and makes it easier for visitors to find the book they want to find. Presentation of information on books available in the library of AMIK Stiekom North Sumatra has not been effective and efficient, because it is still done by looking at the library employee records on the list of books that have been provided. This problem becomes one of the factors to decrease interest in reading the library visitors. Utilization of application suggestions reading books in the library of AMIK Stiekom North Sumatra can be used as a trigger of the interest of visitors to keep reading books available in the library. The built-in book suggestion app works based on brute-force algorithms so that it can present book information relevant to the topic being searched by the visitors.

Keywords : *Library, Applications, Search, Book, Brute Force*

Abstrak

Layanan informasi tentang buku-buku pada suatu perpustakaan sangat membantu dan mempermudah para pengunjung untuk menemukan buku yang ingin dicari. Penyajian informasi buku-buku yang tersedia pada perpustakaan AMIK Stiekom Sumatera Utara selama ini masih belum efektif dan efisien, karena masih dilakukan dengan melihat catatan-catatan pegawai perpustakaan pada daftar buku yang telah disediakan. Masalah ini justru menjadi salah satu faktor menurunkan minat baca para pengunjung perpustakaan. Pemanfaatan aplikasi saran buku bacaan di perpustakaan AMIK Stiekom Sumatera Utara dapat dijadikan sebagai salah pemicu minat para pengunjung untuk tetap membaca buku-buku yang tersedia di perpustakaan. Aplikasi saran buku yang dibangun bekerja berdasarkan algoritma brute force sehingga dapat menyajikan informasi buku yang relevan dengan topik yang sedang dicari oleh para pengunjung.

Kata Kunci: *Perpustakaan, Aplikasi, Pencarian, Buku, Brute Force*

1. PENDAHULUAN

Amik Stiekom Sumatera Utara merupakan salah satu Perguruan Tinggi yang ada di wilayah kerja Koordinator Perguruan Tinggi Swasta (Kopertis) Wilayah I yang sejak awal telah memiliki ruangan khusus sebagai perpustakaan. Perpustakaan Amik Stiekom Sumatera Utara telah menyediakan berbagai buku-buku pendukung yang dapat dipelajari dan digunakan oleh mahasiswa sebagai referensi bagi matakuliah. Secara umum, buku-buku yang tersedia di perpustakaan merupakan buku-buku ilmu komputer terutama buku yang mendukung matakuliah yang diajarkan di Amik Stiekom Sumatera Utara. Buku-buku tersebut disusun dan dikelompokkan berdasarkan judulnya kemudian diletakkan pada rak yang telah tersedia di dalam ruangan perpustakaan.

Hingga saat ini, di Amik Stiekom Sumatera Utara belum memiliki aplikasi yang mampu memberikan layanan informasi mengenai judul-judul buku yang disarankan

untuk dibaca oleh pengunjung perpustakaan khususnya para mahasiswa sesuai dengan topik yang sedang dicari. Namun petugas perpustakaan hanya dapat memberikan informasi tentang keberadaan buku (posisi rak buku) yang sedang dicari oleh mahasiswa berdasarkan catatan daftar buku yang tersedia di perpustakaan.

Berdasarkan penelitian yang dilakukan oleh Puspa, mengatakan bahwa sebuah perpustakaan haruslah menyediakan aplikasi yang mampu memberikan layanan informasi tentang buku perpustakaan bagi para pengunjung[1], [2]. Sedangkan penelitian lain mengatakan layanan perpustakaan dilakukan semata-mata untuk memberikan kepuasan bagi pengguna, oleh karena itu perpustakaan membutuhkan penyusunan katalog buku yang baik sehingga dapat memudahkan proses pencarian serta penyajian informasi yang dibutuhkan[3].

Algoritma *brute force* merupakan salah satu algoritma yang dapat dimanfaatkan untuk melakukan pencarian berdasarkan pencocokan *string*. Algoritma *brute force* bekerja dengan membandingkan *string* yang diinput oleh pengguna dengan semua *string* yang ada hingga ditemukannya *string* yang menjadi *string* target pencarian[4].

Penelitian ini menguraikan bagaimana merancang dan membangun sebuah aplikasi perpustakaan yang mampu memberikan saran judul buku bacaan bagi pengunjung. Aplikasi yang dibangun akan bekerja berdasarkan algoritma *brute force* untuk mencari judul buku yang relevan dengan topik yang sedang dicari. Hal ini dapat membantu petugas perpustakaan dalam memberikan layanan yang optimal serta mampu meningkatkan minat baca para pengunjung.

2. METODOLOGI PENELITIAN

2.1. Perpustakaan

Institusi sebagai pengelola koleksi karya tulis, karya cetak, dan atau karya rekam secara profesional guna memenuhi kebutuhan pendidikan, penelitian, pelestarian, informasi, dan rekreasi para pemustaka merupakan definisi dari perpustakaan[3]. Sebuah perpustakaan dapat menggunakan mekanisme tertentu untuk melaksanakan fungsinya untuk memberi pelayanan terhadap pengelolaan dan pemeliharaan serta penyampaian informasi mengenai karya tulis melalui mekanisme dan sistem tertentu.

2.2. Layanan Perpustakaan

Tugas utama perpustakaan khusus adalah menyediakan layanan informasi yang mencakup: layanan referensi, layanan penelusuran dan temu kembali informasi, layanan sirkulasi, konsultasi bibliografi, penyediaan fasilitas membaca/belajar yang nyaman. Selain itu perpustakaan harus pula menyediakan layanan pendidikan pemakai untuk meningkatkan wawasan dan keterampilan pemustaka dalam menggunakan layanan, koleksi, dan fasilitas pendukung perpustakaan, khususnya pemanfaatan teknologi informasi dan komunikasi[1], [3].

Barometer keberhasilan perpustakaan adalah dapat memberikan pelayanan yang berkualitas, tepat guna, hal ini sangat dipengaruhi oleh prosedur prosedur didalam perpustakaan tersebut sehingga dapat memberikan pelayanan terbaik. Pelayanan yang baik, cepat dan tepat memberi nilai tambah bagi pemustaka yang

berkunjung ke perpustakaan, pelayanan dikatakan baik apabila dilakukan dengan cepat, tepat waktu dan benar.

2.3. String Matching

Pencocokan *string* merupakan bagian penting dari sebuah proses pencarian *string* (*string searching*) dalam sebuah dokumen. Hasil dari pencarian sebuah *string* dalam dokumen tergantung dari teknik atau cara pencocokan *string* yang digunakan. Agar isi dokumen yang benar sesuai dengan kebutuhan informasi dapat diketahui dengan benar, maka dapat diperlukan metode pencarian *string* (*string searching*) isi dokumen yang bagus. Proses pencocokan *string* (*string matching*) yang merupakan bagian utama dalam proses pencarian *string* memegang peranan penting untuk mendapatkan dokumen yang sesuai dengan kebutuhan informasi tersebut. *String Matching* merupakan salah satu algoritma yang digunakan untuk mempercepat proses pencarian kata yang diinginkan. *String matching* dibagi menjadi dua, yakni *exact matching* dan *heuristic* atau *statistical matching*[5], [6].

2.4. Algoritma Brute Force

Algoritma *brute force* merupakan algoritma yang digunakan untuk mencocokkan *pattern* dengan semua *string* antara 0 dan n-m untuk menemukan keberadaan *string target*. pemecahan masalah yang sangat sederhana, langsung, dan jelas dapat dilakukan berdasarkan algoritma ini. Bila si penyusun algoritma lebih mempertimbangkan untuk memperoleh solusi dari *problem* secara langsung apa adanya maka algoritma *brute force* sangat tepat untuk digunakan[7]. Secara rinci langkah-langkah algoritma *brute force* untuk mencocokkan string [8] adalah:

1. Pencocokan *patern* dimulai mulai dari awal teks.
2. Algoritma *brute force* akan mencocokkan karakter per karakter *pattern* dengan karakter pada teks yang bersesuaian yang dimulai dari kiri ke kanan, sampai salah satu kondisi berikut terpenuhi :
 - a. Karakter di *pattern* yang dibandingkan cocok, maka pencarian selesai.
 - b. Apabila dijumpai ketidakcocokan antara *pattern* dengan teks, maka pencarian tidak cocok dan belum berhasil.
3. Kemudian algoritma *brute force* terus menggeser *pattern* sebesar satu ke kanan, dan mengulangi langkah ke-2 sampai *pattern* berada di ujung teks. Seperti halnya dengan algoritma lainnya, algoritma *brute force* juga mempunyai kelebihan dan kekurangan.

Kelebihan *brute force* adalah :

1. Algoritma *brute force* dapat digunakan untuk memecahkan hampir dari sebagian besar masalah.
2. Algoritma *brute force* sederhana dan mudah dimengerti.
3. Algoritma *brute force* menghasilkan algoritma yang layak untuk beberapa masalah penting seperti pencarian, pengurutan, pencocokkan *string*, atau perkalian matriks
4. Algoritma *brute force* menghasilkan algoritma baku (standart) untuk tugas-tugas komputasi penjumlahan/perkalian n buah bilangan, menentukan elemen minimum atau maksimum di dalam tabel (*list*)

Kelemahan dari algoritma *brute force* adalah tidak sekonstruktif teknik pemecahan masalah lainnya.

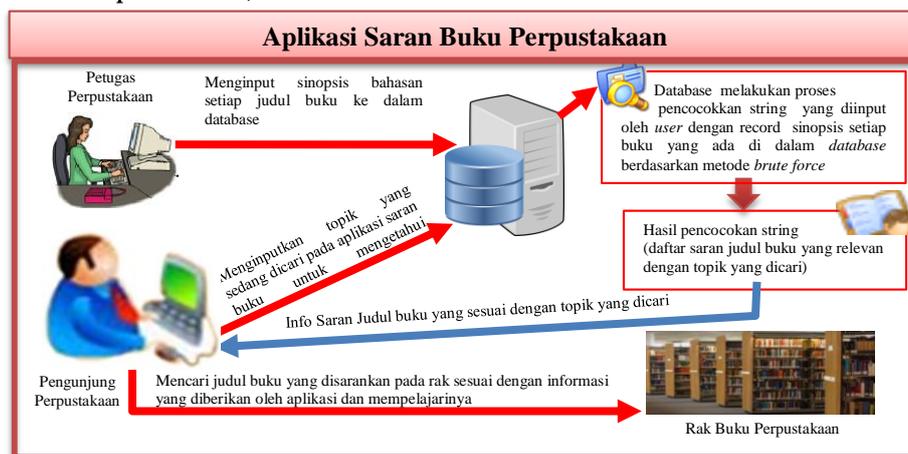
3. HASIL DAN PEMBAHASAN

3.1. Analisa Masalah

Pemberian layanan yang optimal kepada pengunjung perpustakaan merupakan salah satu hal yang harus diwujudkan oleh setiap perpustakaan. Layanan yang umum dibutuhkan oleh para pengunjung adalah layanan informasi mengenai buku yang sedang dicari. Tidak optimalnya layanan seperti ini dapat memberikan efek terhadap penurunan minat pengunjung untuk membaca buku-buku yang tersedia di dalam perpustakaan. Masalah inilah yang terjadi di perpustakaan AMIK Stiekom Sumatera Utara saat ini.

Salah satu solusi yang dapat dilakukan untuk mengefektifkan layanan perpustakaan kepada pengunjung adalah pemanfaatan sebuah aplikasi saran buku bacaan yang mampu menyajikan informasi buku yang disarankan untuk dibaca oleh para pengunjung berdasarkan kata kunci buku yang dicari. Agar proses pencarian lebih efektif dan akurat, maka proses pencarian tersebut dilakukan berdasarkan algoritma *brute force* yang merupakan salah satu dari algoritma teknik pencocokan *string* (*string matching*).

Algoritma *brute force* difungsikan untuk melakukan proses pencarian buku yang disarankan kepada pengunjung yang relevan dengan kata kunci yang diinput oleh pengunjung. Algoritma *brute force* melakukan pencarian buku yang disarankan kepada pengunjung dengan menelusuri data sinopsis pembahasan masing-masing buku. Bila kata kunci pencarian yang diinput oleh pengunjung terdapat di dalam *record field* sinopsis buku, maka informasi buku tersebut disarankan untuk dibaca.



Gambar 2. Skema Aplikasi Saran Buku Bacaan

3.2. Implementasi

Aplikasi yang dibangun menggunakan bahasan pemrograman visual basic 2008 dan aplikasi database MySQL[9], [10]. Penyajian informasi buku perpustakaan yang disarankan kepada para pengunjung dilakukan berdasarkan mekanisme pencocokan *string input* dari pengunjung. *String input* ini akan dijadikan sebagai kata kunci pencarian pada *record field* sinopsis buku yang tersedia pada perustakaan AMIK Stiekom Sumatera Utara.

Implementasi algoritma *brute force* dalam proses pencarian informasi buku yang disarankan untuk dibaca oleh pengunjung diasumsikan dilakukan pada *record* tabel *database* di bawah ini.

Tabel 1. Record Tabel Buku

id	Judul	Pengarang	Sinopsis_Buku	Rak
Simbad01	Database Design	Indrajani	Database, ERD, DFD	1A
Algo01	Algoritma dan Pemrograman	Drs. Lamhot Sitorus	Logika, Algoritma, Flowchart, Variabel	1B
Sifo02	Analisis Sistem Informasi	Tata Sutabri	Sistem, Informasi, DFD, Database	2C

Proses penelusuran kata sinopsis buku yang sama dengan kata kunci yang diinput oleh pengunjung perpustakaan dilakukan pada setiap *record* field sinopsis buku. Bila *record* yang ditelusuri mengandung kata kunci, maka buku pada *record* tersebut akan disarankan untuk dibaca oleh pengunjung. Bila diasumsikan kata kunci yang diinput oleh pengunjung adalah “**DFD**”, maka proses penelusuran *field* sinopsis buku berdasarkan algoritma *brute force* adalah :
 Penelusuran pada *record* pertama (id Simbad01).

Tahap ke-1																	
Sinopsis Buku	D	a	t	a	b	a	s	E	,		E	R	D	,	D	F	D
String Input	D	F	D														

bandingkan huruf pertama *string input* dengan string pertama sinopsis buku, bila sama maka *string* kedua dari *string input* akan dibandingkan dengan *string* kedua sinopsis buku, bila sama maka *string* ketiga *input* akan dibandingkan lagi dengan *string* ketiga sinopsis, karena *string* ketiga *input* tidak sama dengan *string* ketiga sinopsis buku, maka posisi *string input* digeser ke kanan sebanyak satu kali, kemudian dilakukan proses pencocokan hingga akhir dari *string* sinopsis buku.

Tahap ke-2																	
Sinopsis Buku	D	a	t	a	b	a	s	e	,		E	R	D	,	D	F	D
String Input		D	F	D													

proses pencocokan dilakukan dengan cara yang sama seperti pada tahap ke-1 di atas dan pada tahap kedua *string input* masih belum sama dengan *string* sinopsis buku. Oleh karena itu, maka proses pencocokkan akan diteruskan.

Tahap ke-3																	
Sinopsis Buku	D	a	t	a	b	a	s	e	,		E	R	D	,	D	F	D
String Input			D	F	D												

string input dan *string* sinopsis buku masih belum cocok, maka proses pencocokan diteruskan pada karakter berikutnya.

Tahap ke-4																	
Sinopsis Buku	D	a	t	a	b	a	s	e	,		E	R	D	,	D	F	D
String Input				D	F	D											

Sampai pada tahap keempat masih belum ditemukan kecocokan, maka akan diteruskan proses pencocokan seperti pada tahap sebelumnya, hingga ditemukan kesamaan *string input* dengan *string* sinopsis buku. Hasil pencocokan *string input*

dengan seluruh *string* sinopsis buku (khusus pada *record* pertama), maka ditemukan kesamaan kata kunci “DFD” pada bagian akhir.

Tahap ke-16																
Sinopsis Buku	D	a	t	a	b	a	s	e	,	E	R	D	,	D	F	D
String Input	→												D	F	D	

proses pencocokan pada tahap ke-16, ditemukan bahwa *string input* sama dengan *string* sinopsis buku, maka informasi *record* buku dengan sinopsis tersebut akan disarankan kepada pengunjung dan proses pencocokan pada *record* pertama berhenti dan dilanjutkan pada *record* berikutnya. Berdasarkan proses pencocokan *string input* dengan masing-masing *string record* sinopsis buku, maka didapatkan informasi buku yang disarankan disajikan pada tabel 2 di bawah ini.

Tabel 2. Hasil Pencarian Buku yang Disarankan untuk Dibaca

id	Judul	Pengarang	Sinopsis_Buku	Rak
Simbad01	Database Design	Indrajani	Database, ERD, DFD	1A
Sifo02	Analisis Sistem Informasi	Tata Sutabri	Sistem, Informasi, DFD, Database	2A

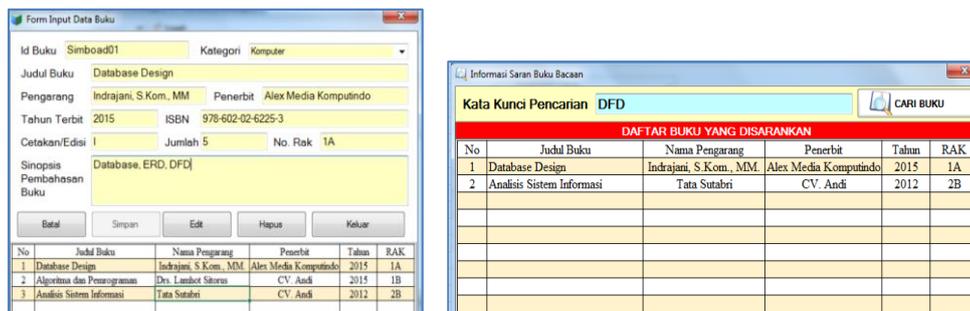
Aplikasi saran buku bacaan yang dibangun terdiri dari empat *form* yaitu *form* menu utama, *form login* admin, *form input* data buku dan *form* pencarian buku.

a. Hasil Implementasi *Form* Menu Utama dan *Login* Admin



Gambar 3. (a) *Form* Menu Utama (b) Implementasi *Form* Login Admin

b. Implementasi *Form* Input Data Buku dan *Form* Pencarian Saran Buku Bacaan



Gambar 4. (a) *Form* Input Data Buku (b) *Form* Saran Buku

4. SIMPULAN

Berdasarkan uraian analisa dan pembahasan, maka disimpulkan beberapa hal sebagai berikut :

- a. Layanan informasi terhadap buku perpustakaan yang disarankan untuk dibaca dapat meningkatkan minat baca pengunjung perpustakaan AMIK Stikom Sumatera Utara.
- b. Pemanfaatan algoritma *brute force* dalam proses pencarian buku bacaan yang disarankan kepada para pengunjung sangat efektif untuk menyajikan informasi saran buku bacaan.
- c. Pencocokan kata kunci pencarian dengan sinopsis buku sangat efektif dalam proses penyajian informasi buku yang disarankan, karena tidak mengharuskan pengunjung untuk menghafal judul buku yang ingin dicari.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih yang sebesar-besarnya kepada DRPM DIKTI yang telah membiayai pelaksanaan penelitian ini dengan skema Penelitian Dosen Pemula (Skema Kompetitif Nasional) yang dituangkan dalam kontrak Nomor 02/LPPM/DIKTI/2018, tanggal 01 Juli 2018.

DAFTAR PUSTAKA

- [1] E. Puspa, "Analisis Kepuasan Pemustaka Terhadap Pelayanan Perpustakaan Pusat Penelitian dan Pengembangan Perikanan Budidaya," *J. Pari*, vol. 2, no. 2, pp. 113–125, 2016.
- [2] T. Zebua and Pritiwanto, "Pembangunan Web Mobile Absensi Mahasiswa Pada Platform Android Yang Terintegrasi Dengan Website Utama Sistem Informasi Akademik (SIA) STMIK Budi Darma," *J. Tek. Inform. Unika St. Thomas*, vol. 2, no. 1, pp. 100–107, 2017.
- [3] F. Nugraha, "Analisa dan Perancangan Sistem Informasi Perpustakaan," *J. SIMETRIS*, vol. 5, no. 1, pp. 27–32, 2014.
- [4] Danuri, "Pencarian File Teks Berbasis Content dengan Pencocokan String Menggunakan Algoritma Brute Force," *Sci. J. Informatics*, vol. 3, no. 1, pp. 68–75, 2016.
- [5] V. Sagita and M. I. Prasetyowati, "Studi Perbandingan Implementasi Algoritma Boyer-Moore, Turbo Boyer-Moore, dan Tuned Boyer-Moore dalam Pencarian String," *ULTIMATICS*, vol. 4, no. 1, pp. 31–37, 2013.
- [6] E. Buulolo, "Implementasi Algoritma String Matching dalam Pencarian Surat dan Ayat dalam Bible Berbasis Android," *Pelita Inform. Budi Darma*, vol. 3, no. 1, pp. 23–27, 2013.
- [7] H. Pratiwi *et al.*, "Implementasi Algoritma Brute Force dalam Aplikasi Kamus Istilah Kesehatan," *J. Ilm. Teknol. Inf. Terap.*, vol. 2, no. 2, pp. 119–125, 2016.
- [8] M. A. Saragih, "Implementasi Algoritma Brute Force dalam Pencocokan Teks Font Italic untuk Kata Microsoft Office Word," *Pelita Inform. Budi Darma*, vol. 4, no. 3, pp. 84–87, 2013.
- [9] T. Sutrabri, *Konsep Sistem Informasi*, I. Jakarta: CV. Andi Publisher, 2012.
- [10] E. Sutanta, *Basis Data Dalam Tunjauan Konseptual*, I. Yogyakarta: Andi, 2011.