

Penerapan Metode *Binary Search* dan *Hamming Distance* pada *E-library* SMAN 2 Katingan Hilir

T K Wulandari¹, E D Oktaviani², A Lestari³

Teknik Informatika, Teknik, Universitas Palangka Raya^{1, 2, 3}

E-mail : theresiakwulandari@gmail.com¹, ennyobrien@gmail.com²,
aries22ta@gmail.com³

Abstrak. Perpustakaan digital (*digital library*) atau disebut juga sebagai perpustakaan *eletronic library (e-library)* diperkenalkan dengan kelebihanannya menyediakan koleksi dalam bentuk elektronik yang mampu diakses oleh pengguna tanpa batas tempat dan waktu. *E-library* yang telah dibuat pada studi kasus SMAN 2 Katingan Hilir ini menerapkan metodologi *waterfall* model Sommerville, dengan beberapa tahapan yaitu *Requirements Analysis and Definition, System and Software design, Implementation and Unit Testing, Integration and System Testing*, dan *Operation and Maintenance*. Beberapa manfaat penggunaan *e-library* ini yaitu pencatatan daftar buku lebih rapi, saat ingin mencari ketersediaan buku lebih cepat, kemudian proses transaksi peminjaman dan pengembalian buku menjadi lebih terkomputerisasi karena melibatkan aplikasi *e-library* dan perangkat komputer yang digunakan. Pada fitur pencarian buku menerapkan algoritma *binary search* dan *hamming distance* dengan tujuan memaksimalkan sistem pencarian. Hasil pengujian dengan *accuracy* 78% dan *recall* 78% tertinggi diperoleh pada skenario pengujian algoritma *hamming distance*, dan algoritma *binary search* diperoleh *accuracy* 72% dan *recall* 72%, dengan hasil yang sama antara dua algoritma *precision* 100%. Dengan adanya *e-library* ini besar harapan untuk meningkatkan minat membaca siswa serta meningkatkan akreditasi bagi sekolah.

Kata Kunci: perpustakaan digital, *e-library*, *Waterfall*, *Binary search*, *Hamming distance*

Abstract. A *digital library (digital library)* is also known as an *electronic library (e-library)* which was introduced with the advantage of providing collections in electronic form that can be accessed by users without limits of place and time. This *e-library* that has been created in the case study of SMAN 2 Katingan Hilir applies the Sommerville model *waterfall methodology*, with several stages, namely *Requirements Analysis and Definition, System and Software design, Implementation and Unit Testing, Integration and System Testing, Operation and Maintenance*. Some of the benefits of using this *e-library* are that the recording of the book list is neater, when you want to find more available books, then the borrowing process and books are faster to become more computerized because they involve the *e-library* application and the computer equipment used. The book search feature applies a *binary search* algorithm and *hamming distance* with the aim of maximizing the search system. The test results with 78% accuracy and the highest 78% recall were obtained in the *hamming distance* algorithm test scenario, and the *binary search* algorithm obtained 72% accuracy and 72% recall, with the same results between the two 100% precision algorithms. With the existence of this *e-library*,

it is hoped that it will increase students' interest in reading and increase accreditation for schools.

Keywords: *digital library, e-library, Waterfall, Binary search, Hamming distance*

1. Pendahuluan

Perpustakaan ialah tempat penyimpanan hasil suatu pemikiran manusia yang hasilnya akan dituangkan ke dalam suatu bahan pustaka baik berupa bentuk cetak maupun non-cetak ataupun dalam bentuk elektronik (digital) [1]. *Electronic library (e-library)* diperkenalkan dengan kelebihanannya menyediakan koleksi dalam bentuk elektronik yang mampu diakses oleh pengguna tanpa batas tempat dan waktu. Jenis perpustakaan ini berbeda dengan jenis perpustakaan konvensional yang berupa kumpulan buku tercetak, film mikro (*microform* dan *microfiche*), ataupun kumpulan kaset audio, video, dan lainnya [2].

Saat ini perpustakaan di SMAN 2 Katingan Hilir masih mengelola perpustakaan secara manual. Perpustakaan SMAN 2 Katingan Hilir saat ini menyimpan buku sebanyak 2.593 berdasarkan jumlah buku dan terdapat 1.124 berdasarkan judul buku. Kondisi perpustakaan saat ini pengelolaan buku, dengan jumlah buku yang terhitung banyak. Dalam pelayanannya pada perpustakaan tidak terlepas dalam satu aktifitas pencarian data. Hal ini disebabkan karena jumlah buku yang terdapat di perpustakaan sudah sangat banyak, sehingga hal ini akan memberikan masalah dalam hal pencarian data buku. Dalam melakukan pencarian data buku perpustakaan, pencocokan *string* merupakan suatu bagian dalam proses pencarian *string (string searching)*. Data buku yang dihasilkan sangat bergantung pada teknik ataupun algoritma yang dilakukan dalam pencocokan *string* tersebut. *Searching* adalah salah satu fitur penting dalam *website*. Fitur ini mempermudah pengguna untuk menemukan apa yang diinginkan tanpa perlu menelusuri setiap halaman yang ada pada suatu *website*. Fitur pencarian ini berkerja dengan cara menerima masukan dari pengguna berupa kata kunci, kemudian kata kunci tersebut diproses dan dicocokkan dengan data *searchable* yang ada di *database website* [3].

Pada penelitian ini, untuk melakukan pencarian digunakanlah algoritma di dalamnya untuk pencarian buku. Untuk mempercepat pencarian buku dan memaksimalkan komputasi, salah satu algoritma dalam proses pencarian adalah algoritma *Binary Search* yaitu algoritma pencarian yang menjelajahi setiap elemen dari tabel. Pada intinya, algoritma ini menggunakan prinsip *divide and conquer*, di mana sebuah masalah atau tujuan diselesaikan dengan cara mempartisi masalah menjadi bagian yang lebih kecil [4].

Dalam proses pencarian menggunakan metode pencarian konvensional, proses selalu mulai melakukan pencarian dari ujung kiri larik di mana hal ini ditunjukkan inisialisasi objek *Index* dengan nilai 0. Langkah pertama algoritma ini adalah menentukan batas bawah (*low*) dan batas atas (*high*), kemudian menentukan titik tengah (*mid*) dengan cara menjumlahkan “batas bawah” (*low*) dan “batas atas” (*high*) dan hasil akhirnya dibagi dengan 2. Dalam matematika, hal ini dapat dituliskan dengan formula $mid = (low + high) / 2$, Jika nilai yang dicari sama dengan batas bawah dan batas atas, berarti data yang dicari telah ditemukan. Jika tidak, berarti data yang dicari memang tidak ada di *arraylist* oleh Robert Lafore [5].

Namun, pada algoritma *binary search* ini memiliki kelemahan dalam kompleksitas pencariannya, untuk mengatasi beberapa pola *user* menginput dalam pencarian, digunakanlah algoritma *hamming distance* untuk mengatasi kelemahan tersebut. [6] *Distance* adalah metode pengoreksian dengan cara membandingkan dua *string* yang panjangnya sama, dan menghitung jarak kesalahan minimal dengan melakukan substitusi sampai diperoleh *string* yang tepat. *Hamming Distance* umumnya digunakan untuk proses *searching* serta *error detecting* dan *error correction code* pada bidang telekomunikasi.

2. Metode

2.1 Metode Pengumpulan Data

a. Observasi

Tahapan ini merupakan pengumpulan data dengan melakukan peninjauan langsung ke SMAN 2 Katingan Hilir untuk mendapatkan data yang aktual.

b. Wawancara

Tahapan ini merupakan pengumpulan data dengan melakukan wawancara untuk mendapatkan persetujuan permintaan data sesuai dengan data yang dibutuhkan melalui kepala sekolah, wakil kepala sekolah dan staf perpustakaan SMAN 2 Katingan Hilir.

c. Studi literatur

Studi literatur dilakukan dengan cara mempelajari referensi-referensi, baik melalui buku-buku, artikel, internet, mengenai perkembangan *e-library* maupun *website* dan memilih fitur-fitur yang akan diterapkan dalam *website e-library*.

d. Metode konsultasi

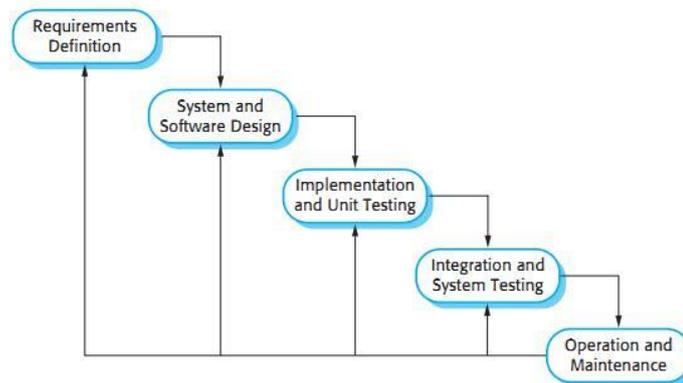
Metode konsultasi ini merupakan proses bimbingan dan diskusi kepada dosen pembimbing mengenai *project* pembuatan *website* serta penyusunan laporan.

e. Metode implementasi

Metode implementasi ini merupakan proses pembuatan *website* dengan data yang telah dikumpulkan.

2.2 Metode Pengembangan Perangkat Lunak

Metode Pengembangan perangkat lunak yang digunakan adalah metode *Waterfall* menurut Sommerville [7]. *Waterfall* model adalah sebuah contoh dari proses perencanaan, di mana semua proses kegiatan harus terlebih dahulu direncanakan dan dijadwalkan.



Gambar 1. *Waterfall Model*

a. *Requirements Definition* (definisi kebutuhan)

Langkah ini merupakan analisa terhadap kebutuhan yang diperlukan dalam pembuatan untuk mendapatkan pilihan dan solusi fitur apa yang akan dirancang.

b. *System and software design* (desain sistem dan perangkat lunak)

Pada tahap ini dilakukan desain aplikasi yang meliputi desain *interface* atau tampilan *website* dengan menerjemahkan sesuai dengan syarat/kebutuhan ke dalam sebuah representasi aplikasi.

c. *Implementation and unit testing* (implementasi dan pengujian unit)

Tahapan inilah merupakan mengerjakan suatu sistem. Di mana desain sistem dan desain *interface* aplikasi yang dirancang sebelumnya diimplementasikan dengan melakukan pembangunan aplikasi yang diterjemahkan ke kode-kode dalam satu set program atau unit program.

Salah satu ukuran jarak antara dua buah untai ialah Jarak *Hamming* (*Hamming Distance*). Nama ini diambil dari nama seorang ilmuwan, yakni Richard *Hamming*. Ia memperkenalkan

ukuran jarak tersebut pada makalahnya yang berjudul *Error detecting and correcting codes* Tahun 1950. Jika diberikan dua buah untai x dan y dengan Panjang yang sama, misalkan n, maka jarak *hamming* antara untai x dan y adalah banyaknya karakter pada untai x yang berbeda dengan karakter pada untai y pada *posisi yang bersesuaian* menurut Hamming R W [8].

Fungsi *HD* didefinisikan sebagai berikut :

$$HD(x, y) = d(x_1, y_1) + d(x_2, y_2) + \dots + d(x_n, y_n) \quad (1)$$

$$\sum_{i=1}^n d(x_i, y_i) \quad (2)$$

Dengan : $x_i, y_i \in V$ untuk $i = 1, 2, \dots, n$

$$d(x_i, y_i) = 0 \text{ jika } x_i = y_i \text{ dan} \quad (4)$$

$$d(x_i, y_i) = 1 \text{ jika } x_i \neq y_i \quad (5)$$

d. *Integration and system testing* (integrasi dan pengujian sistem)

Dalam tahapan ini, setiap unit program akan diintegrasikan satu sama lain atau menyatukan semua unit program untuk diuji secara keseluruhan untuk mendeteksi apakah ada *bug* atau *error* di dalam *website* ini agar terjamin bahwa persyaratan sistem telah dipenuhi.

Menurut Han dan Kamber [9] *confusion matrix* adalah alat yang berguna untuk menganalisis seberapa baik *classifier* mengenali *tuple* dari kelas yang berbeda.

		Predicted class		Total
		yes	no	
Actual class	yes	TP	FN	P
	no	FP	TN	N
Total		P'	N'	P + N

Gambar 2. *Confusion matrix*

Untuk mengukur performa klasifikasi menggunakan *confusion matrix* dapat dilakukan dengan beberapa hal yaitu :

a. *Accuracy*

Akurasi merupakan rasio prediksi benar (positif dan negatif) dengan keseluruhan data.

$$Akurasi = \frac{TP+TN}{TP+FP+FN+TN} \quad (6)$$

b. *Precision*

Precision merupakan rasio prediksi benar positif dibandingkan dengan keseluruhan hasil yang diprediksi positif.

$$Precision = \frac{TP}{TP+FP} \quad (7)$$

c. *Recall*

Recall merupakan rasio prediksi benar positif dibandingkan dengan keseluruhan data yang benar positif.

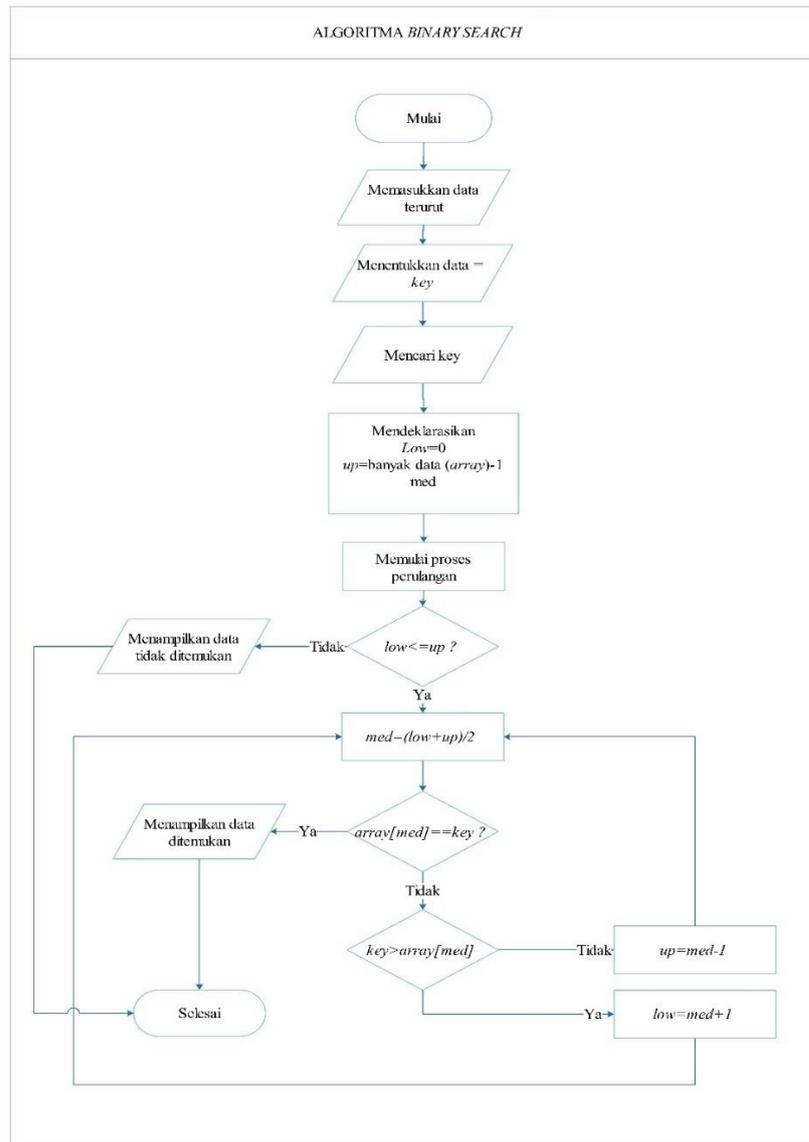
$$Recall = \frac{TP}{TP+FN} \quad (8)$$

e. *Operation and maintenance* (operasi dan pemeliharaan)

Pada tahapan ini sistem dipasang atau mulai digunakan. Melakukan juga pemeliharaan yang mencakup koreksi dan berbagai kesalahan yang tidak ditemukan pada tahap-tahap sebelumnya

3. Hasil dan Pembahasan

3.1 Flowchart Pencarian Algoritma Binary Search

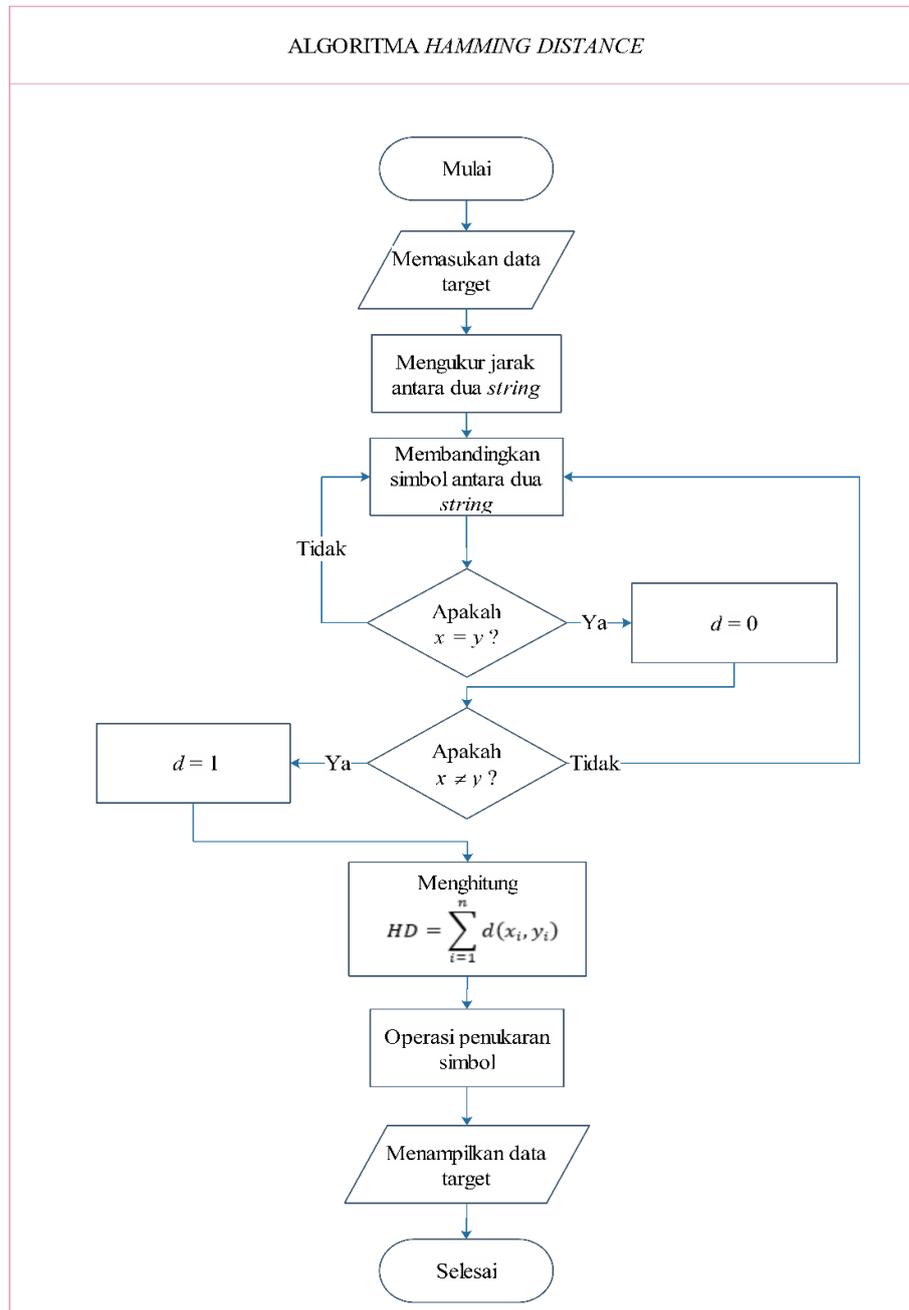


Gambar 3. Flowchart Proses Pencarian Algoritma Binary Search

Algoritma *Binary Search* yaitu algoritma pencarian yang menjelajahi setiap elemen dari tabel. Pada intinya, algoritma ini menggunakan prinsip *divide and conquer*, di mana sebuah masalah atau tujuan diselesaikan dengan cara mempartisi masalah menjadi bagian yang lebih kecil [5]. Langkah pertama algoritma ini adalah menentukan batas bawah (*low*) dan batas atas (*high*), kemudian menentukan titik tengah (*mid*) dengan cara menjumlahkan “batas bawah” (*low*) dan “batas atas” (*high*) dan hasil akhirnya dibagi dengan 2. Dalam matematika, hal ini dapat dituliskan dengan formula $mid = (low + high) / 2$. Jika nilai yang dicari lebih kecil dari titik tengah

(*mid*), batas atas (*high*) dihitung dengan mengurangi sebesar 1 posisi dari titik tengah dan sebaliknya. Algoritma *binary searching* ini akan berulang hingga batas bawah (*low*) sama dengan batas atas (*high*), yang berarti pencarian pada *arraylist* sudah berakhir.

3.2. Flowchart Pencarian Algoritma *Hamming Distance*

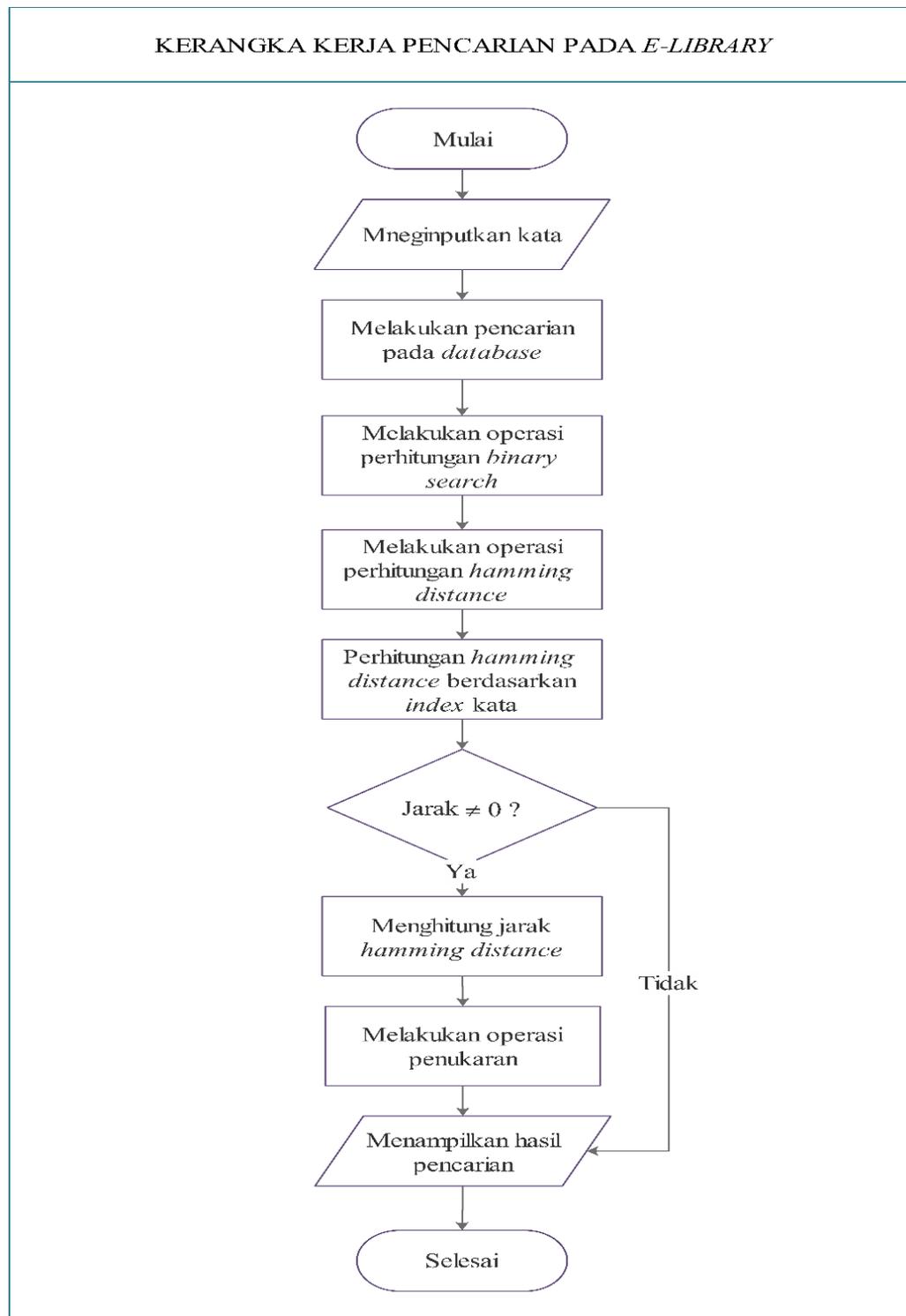


Gambar 4. Flowchart Proses Pencarian Algoritma *Hamming Distance*

Hamming Distance adalah metode pengoreksian dengan cara membandingkan dua *string* yang panjangnya sama, dan menghitung jarak kesalahan minimal dengan melakukan substitusi. *Hamming Distance* umumnya digunakan untuk proses *searching* serta *error detecting* dan *error correction code* pada bidang telekomunikasi. Dalam hal ini *hamming distance* melakukan pengkoreksian dengan beberapa tahap yaitu membandingkan 2 *string* dan menghitung jarak

antara dua buah *string* kemudian men-substitusi karakter yang berbeda hingga memperoleh *string* yang tepat.

3.3. Flowchart Kerangka Kerja Pencarian pada E-library



Gambar 5. Flowchart Kerangka kerja Pencarian pada E-library

Mekanisme pencarian pada *e-library* dilakukan dengan memasukkan kata yang kemudian sistem akan melakukan pencarian pada *database* dengan membandingkan kata inputan dan kata

yang ada pada *database*. Selanjutnya, sistem operasi menjalankan perhitungan *binary search* berdasarkan *index* data buku dan operasi perhitungan *hamming distance* berdasarkan *index* kata dengan menghitung jarak *hamming distance*. Jika jarak $\neq 0$, sistem akan melakukan operasi penukaran pada kata inputan dengan kata yang berada pada *database*. Setelah data ditemukan, sistem akan menampilkan pencarian.

3.4. Pengujian Algoritma Pencarian

a. Skenario Pencarian Menggunakan Algoritma *Binary Search*

Skenario pengujian pencarian judul buku menggunakan algoritma pencarian *binary search* pada pengujian ini menggunakan 50 kata yang diambil secara acak. Contoh kata-kata ditunjukkan pada Tabel 1.

Tabel 1. Tabel Skenario Pencarian Algoritma *Binary Search*

No	Kata untuk Pencarian Algoritma <i>Binary Search</i>	Jumlah Kata yang Diuji	Jumlah Kata yang di Temukan	Menemukan Kata
1	Inggris	22	22	Ada
2	Bahasa Inggris	22	22	Ada
3	Matematika	70	70	Ada
4	Buku Matematika	70	0	tidak ada
5	Agama	60	60	Ada
6	Cerita Rakyat	3	3	Ada
7	Kamus	50	50	Ada
8	Kamus Indonesia	39	0	tidak ada
9	Pendidikan	36	36	ada
10	Budi daya ikan	1	0	tidak ada

Dari hasil pengujian di atas perhitungan akurasi, *precision*, dan *recall* rasio kata yang dapat ditemukan pada pencarian judul buku di *website* (positif dan negatif) dengan keseluruhan data. Dapat diketahui akurasi pada algoritma *binary search* dengan jumlah kata yang diuji sebanyak 50 kata dengan 36 kata ditemukan dan 14 kata tidak ditemukan, nilai yang dihasilkan sebesar akurasi 72%, *precision* 100%, *recall* 72%.

Tabel 2. Tabel *Confussion Matrix* Pengujian *binary search*

	1 (Positive)	0 (Negative)
1 (Positive)	TP 36	FP 0
0 (Negative)	FN 14	TN 0
	50	0

$$\text{Akurasi} = \frac{36+0}{36+0+0+14} = \frac{36}{50} \times 100\% = 72\% \quad (9)$$

$$\text{Precision} = \frac{36}{36+0} = \frac{36}{36} \times 100\% = 100\% \quad (10)$$

$$\text{Recall} = \frac{36}{36+14} = \frac{36}{50} \times 100\% = 72\% \quad (11)$$

b. Skenario Pencarian Menggunakan Algoritma *Hamming Distance*

Skenario pengujian pencarian judul buku menggunakan algoritma *hamming distance* pada pengujian ini menggunakan 50 kata yang diambil secara acak. Contoh kata-kata ditunjukkan pada Tabel 3.

Tabel 3. Tabel Skenario Pencarian Algoritma *Hamming Distance*

No	Kata untuk Pencarian Algoritma <i>Hamming Distance</i>	Kata yang Benar	Jumlah Kata yang Diuji	Jumlah Kata yang ditemukan	Selisih Karakter	Menemukan Kata
1	Kmus Pratis Mdern Bahasa Indoesia	Kamus Praktis Modern Bahasa Indonesia	2	2	5	Ada
2	Katolik	Khatolik	1	1	1	Ada
3	Khatolik	Katolik	1	1	1	Ada
4	Perempuan	Peyempuan	1	0	1	tidak ada
5	You Knw I Lope Yu But	You Know I Love You But	1	1	3	ada
6	Kasiat Kacang Gude	Khasiat Kacang Gude	1	1	1	ada
7	Khasiat Kacang Gude	Khasiat Kacang Gude	1	1	1	ada
8	Dasyatnya Doa Ank Yatm	Dahsyatnya Doa Anak Yatim	1	1	3	ada
9	Kamus Pntn Indonesia	Kamus Pantun Indonesia	1	1	2	ada
10	Cerita Tntang Cnta	Cerita Tentang Cinta	1	1	2	ada

Dari hasil pengujian di atas perhitungan akurasi, *precision*, *recall* rasio kata yang dapat ditemukan pada pencarian judul buku di *website* (positif dan negatif) dengan keseluruhan data. Dapat diketahui akurasi pada algoritma *hamming distance* dengan jumlah kata yang diuji sebanyak 50 kata dengan 38 kata ditemukan dan 11 kata tidak ditemukan, nilai yang dihasilkan sebesar akurasi 78%, *precision* 100%, *recall* 78%.

Tabel 4. Tabel *Confussion Matrix* Pengujian *binary search*

	1 (Positive)	0 (Negative)
1 (Positive)	TP 39	FP 0
0 (Negative)	FN 11	TN 0
	50	0

$$\text{Akurasi} = \frac{39+0}{39+0+0+11} = \frac{39}{50} \times 100\% = 78\% \quad (12)$$

$$\text{Precision} = \frac{39}{39+0} = \frac{39}{39} \times 100\% = 100\% \quad (13)$$

$$\text{Recall} = \frac{39}{39+11} = \frac{39}{50} \times 100\% = 78\% \quad (14)$$

4. Kesimpulan

Kesimpulan dari tugas akhir yang berjudul “Penerapan Metode *binary search* dan *hamming distance* pada *e-library* SMAN 2 Katingan Hilir” adalah sebagai berikut: *e-library* yang telah dibuat pada studi kasus SMAN 2 Katingan Hilir ini menerapkan metodologi *waterfall* model Sommerville. Pada fitur pencarian buku menerapkan algoritma *binary search* dan *hamming distance* dengan tujuan memaksimalkan sistem pencarian serta komputasinya yang dapat mengatasi beberapa pola *user* memasukan inputan. Hasil pengujian dengan *blackbox testing* menunjukkan sistem dapat berjalan dengan baik. Pengujian dengan metode *confusion matrix* yaitu dengan *accuracy* 78% dan *recall* 78% tertinggi diperoleh pada skenario pengujian algoritma *hamming distance*, dan algoritma *binary search* diperoleh *accuracy* 72% dan *recall* 72%, dengan hasil yang sama antara dua algoritma *precision* 100%. Dengan demikian, *website e-library* ini dapat meningkatkan pelayanan pada perpustakaan SMAN 2 Katingan Hilir.

Referensi

- [1] Suwarno, Wiji. 2010. *Pengetahuan Dasar kepustakaan*. Bogor : Ghalia Indonesia
- [2] Anhar. 2015. *Perancangan Aplikasi Pengambilan Keputusan Penentuan Paket Tujuan Wisata Menggunakan Metode Topsis*, Informatika dan Teknologi Ilmiah (INTI) Vol V.
- [3] Oktaviyani, E. D., Christina, S., & Ronaldo, D. (2019, December). Keywords Search Correction Using Damerau Levenshtein Distance Algorithm. In Conference SENATIK STT Adisutjipto Yogyakarta (Vol. 5, pp. 167-176).
- [4] Munir, Rinaldi. 2011. *Algoritma dan Pemrograman*. Bandung : Informatika.
- [5] Munir, Rinaldi. 2011. *Algoritma dan Pemrograman dalam Bahasa Pascal dan C Edisi Revisi*. Bandung: Informatika Bandung.
- [6] Wardhana, Wedha Satya, dan Wirayuda, Tjokorda Agung Budi, dan Shaufiah. 2011. *Pengoreksian Ejaan Kata Menggunakan Metode N-Gram (Studi Kasus Dokumen Teks Berbahasa Indonesia)*. Bandung : TELKOM University
- [7] Sommerville, Ian. 2011. *Software Engineering. 9th Edition*. America : Pearson. Education, Inc.
- [8] Fadillah, Muhamad Rafly. 2011. *Pencocokan untai dengan menggunakan ukuran jarak hamming*. Jakarta : Universitas Indonesia
- [9] Han, J., Kamber, M., & Pei, J. 2011. *Data Mining Concepts and Techniques Third Edition*. Waltham: Elsevier Inc.
- [10] Suzamkonyden & Wijaya, Mikye. 2012. *Evaluasi sistem informasi perpustakaan UIN Suska berdasarkan Organisasi, Manajemen dan Teknologi*. Pekanbaru: Fakultas Sains dan Teknologi Riau. Yogyakarta : Andi.