

# Perbandingan Efisiensi Algoritma String Matching Knuth Morris Pratt Dan Algoritma Levenshtein Pada Aplikasi Pengarsipan Dan Pencarian Data Anggota Honda Megapro Club Indonesia

Nurul Huda<sup>\*1)</sup>, Muhammad Khafidurrohman<sup>\*2)</sup>

<sup>1</sup>Program Studi Teknik Informatika, Universitas Bina Darma Palembang,  
Jl. Jenderal A. Yani No.3, 9/10 Ulu, Seberang Ulu I, Palembang, Sumatera Selatan 30111  
e-mail: [\\*nurul\\_huda@binadarma.ac.id](mailto:nurul_huda@binadarma.ac.id), [\\*mhd.khafi26@gmail.com](mailto:mhd.khafi26@gmail.com),

## Abstrak

*Kebutuhan pemeliharaan data sangat banyak terlebih lagi pada data yang bersifat sangat penting, namun terkadang dalam melakukan pencarian data terdapat beberapa kendala. Beberapa diantaranya kesalahan pada penulisan text pencarian, yang menyebabkan pencarian data memakan waktu yang tidak sedikit dan bahkan membuat data tidak di temukan. Beberapa metode pencarian data yaitu KMP maupun Leveshtein yang memiliki keunggulan tersendiri. Pada penelitian ini penulis bertujuan melakukan perbandingan antara metode KMP dan Levenshtein dengan menggunakan metode penelitian Eksperimen, untuk mengetahui hasil dari metode pencarian data mana yang cara kerja dan proses pencariannya lebih cepat, agar memberikan manfaat bagi pengguna untuk mempermudah dalam melakukan pencarian data. Hasil penelitian menunjukkan bahwa kinerja yang dimiliki algoritma levenshtein lebih unggul di bandingkan dengan algoritma knuth morris pratt, dengan hasil kecepatan dan ketepatan dalam melakukan pencarian data anggota yaitu 0,022 untuk rata-rata pencarian algoritma knuth morris pratt dan 0,02 untuk rata-rata pencarian algoritma levenshtein.*

**Kata kunci**—Perbandingan, Knuth Morris Pratt (KMP), Levenshtein.

## Abstract

*The need for data maintenance is very large, especially for data that is very important, but sometimes in searching for data there are several obstacles. Some of them were errors in writing the search text, which caused the search for data to take a lot of time and even made the data not found. Several data search methods, namely KMP and Leveshtein, have their own advantages. In this study the author aims to make a comparison between the KMP and Levenshtein methods using the Experimental research method, to find out the results of which data search method works and the search process is faster, in order to provide benefits for users to make it easier to search data. The results of the study show that the performance of the Levenshtein algorithm is superior compared to the Knuth Morris Pratt algorithm, with the result that the speed and accuracy in searching for member data is 0.022 for the average search algorithm for Knuth Morris Pratt and 0.02 for the average search algorithm. levenshtein.*

**Keywords**— Comparison, Knuth Morris Pratt (KMP), Levenshtein.

## 1. PENDAHULUAN

Kebutuhan pemeliharaan data tidak dapat di hindari baik dalam ruang lingkup individu, organisasi, perusahaan maupun pemerintahan. Namun terkadang dalam pemeliharaan data mengalami beberapa kendala beberapa diantaranya yaitu pencarian data.

Honda Megapro Club Indonesia (HMPCI) merupakan suatu organisasi kendaraan roda

dua yang menjadi salah satu pelopor keselamatan berkendara di kalangan masyarakat, organisasi tersebut pertamakali berdiri di kota Jakarta, 06 Desember 2003 hingga saat ini, seiring berjalannya waktu organisasi tersebut mampu memperlebar sayapnya dari sabang sampai merauke[1].

Mengenal banyaknya anggota yang terdaftar di organisasi tersebut timbul suatu kendala dalam pemeliharaan dan pencarian data yang belum efektif. Dalam melakukan pencarian dapat menggunakan sistem yang telah tersedia. Sebagaimana contoh dengan menegtik kata kunci “kafi” padahal ejaan baku dalam sistem tersebut adalah “khafi”. Tentu akan menyebabkan pencarian data memakan waktu yang tidak sedikit dan bahkan membuat data tidak di temukan.

Algoritma untuk implementasi pencarian data terus mengalami perkembangan diantaranya algoritma pencarian string atau *string matching*. Algoritma *knuth morris pratt* adalah suatu algoritma pencarian string yang dikembangkan terpisah oleh James H. Morris bersama Vaughan R. Pratt dan Donald E. Knuth[2]. Algoritma *levenshtein* atau disebut juga *levenshtein distance* adalah algoritma pencarian jumlah perbedaan string yang ditemukan oleh Vladimir Levenshtein[3]. Ilham dan Mirza menyatakan bahwa Algoritma KMP sudah cukup cepat dan optimal dalam menemukan kata yang dicari didalam sebuah kalimat[4]. Sumiari dan Iis mengemukakan bahwa Akurasi algoritma *levenshtein* dalam optimasi pencarian data cukup optimal[5].

Dari penjelasan diatas penulis tertarik untuk melakukan perbandingan performasi proses pencarian data pada Algoritma *String matching Knuth Morris Pratt* (KMP) dengan Algoritma *Leveshtein* pada aplikasi pengarsipan dan pencarian data anggota Honda Megapro Club Indonesia (HMPCI) untuk menghasilkan informasi data yang sesuai dengan keinginan pengguna..

Penelitian sebelumnya terkait antara dua algoritma tersebut oleh Nababan dan Jannah di tahun 2019 yang berjudul “Algoritma String Matching Brute Force dan Knuth Morris Pratt Sebagai Search Engine Berbasis Web Pada Kamus Istilah Jaringan Komputer” Nababan dan Jannah melakukan penelitian yang menjelaskan hasil bahwa Algoritma *Brute Force* dan *Algoritma Knuth Morris Pratt* telah di terapkan dengan baik, namun Algoritma *Knuth Morris Pratt* melakukan pengecekan string yang lebih efisien dibandingkan dengan Algoritma *Brute Force*. Algoritma *Knuth Morris Pratt* memperoleh hasil rata-rata lebih cepat 0,0037 ms dibandingkan dengan algoritma *Brute Force* 0,00399 ms dalam pencocokan string daeri segi *runtime*[6].

Ada juga penelien yang dilakukan oleh Qurnia dkk pada tahun 2022 berjudul Kamus Digital “Tanaman Obat Menggunakan Algoritme Knuth Morris Pratt Berbasis Mobile”. Qurnia dkk telah berhasil mengimplementasikan Algoritma *Knuth Morris Pratt* kinerja sistem dalam pencarian pencocokan menghasilkan nilai yang baik. Dengan hasil evaluasi menggunakan 640 data tanaman dan di uji dengan 30 kata kunci di dapatkan nilai *accuracy* 100%, nilai *recall* 100% dan rata- rata *precision* 92.4%[7].

Penelitian lainnya yang terkait algoritma tersebut dilakukan oleh Islamiyati dan Fikti tahun 2022 dengan judul “Penerapan Algoritma Knuth Morris Pratt dalam Mendeteksi Tingkat Kemiripan Judul Skripsi Berbasis Web”. Dengan hasil analisa judul yang telah di ajukan dan juga dengan proses pencarian data antara *srting* dengan *pattern* dimana hasil yang di ditampilkan oleh sisitem berupa jumlah kata berikut persentase kemiripannya[8].

Dipenelitian lainnya yang ddilakukan oleh Natalie dkk tahun 2023 dengan judul “Optical Character Recognition Menggunakan Uipath Dan Pencocokan Pencocokan Data Sertifikat Dengan Algoritma *Levenshtein Diatance*” dengan hasil penelitian penggunaan algoritma *levenshtein* sangat baik di level sintaktik dalam pengenalan kata pada setifikat berdasatkan hasil pengujian RPA, dengan akurasi sebesar 83,52% dengan waktu RPA berjalan 94,7 ms[9].

Di tahun 2020 Kuswanto melakukan penelitian dengan judul “Implementasi Algoritma *Levenshtein Distance* Dengan Restful Web Service Pada Kata Bahasa Indonesia Ke Bahasa Jawa Berbasis Web”. Kuswanto berhasil mengimplementasikan algoritma *levenshtein*

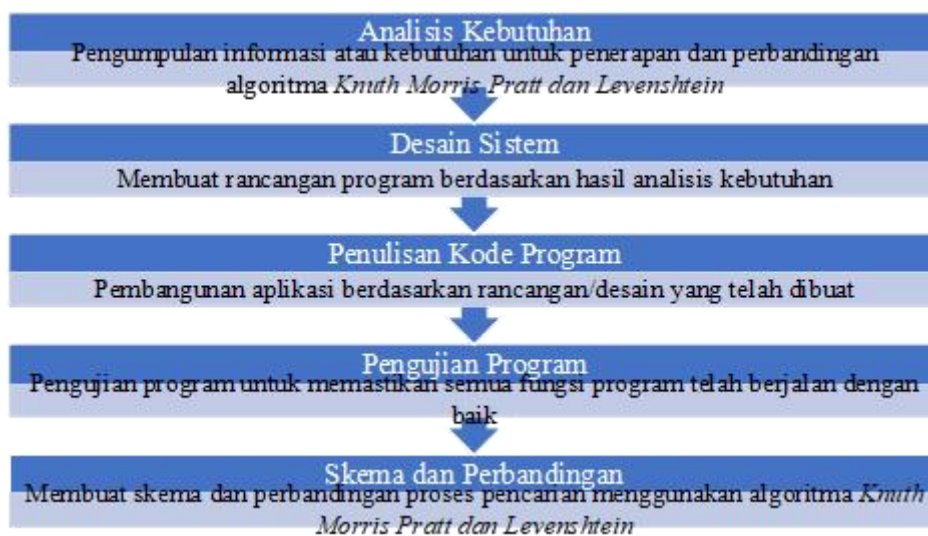
*distance* dengan *resrful web*. Dengan uji coba data sebanyak 30.382 kosakata dan sejumlah 172 data yang ada di database. penulis melakukan pengujian sebanyak duabelas *query*, dengan hasil persentase yang di dapat yaitu nilai *f-measure* 97,51%, nilai *recall* 100%, nilai *accuracy* 99,97%, dan nilai *precision* sebesar 95%. Tidak terdapatb banyak kesalahan proses, sehingga secara fungsional jelas mamapu menampilkan hasil yang begitu sesuai harapan[10].

Penelitian sebelumnya yang di lakukan oleh Manuaba dkk di tahun 2022 yang berjudul “Analisis Setimen Data Provider Layanan Internet Pada Twiter Menggunakan Support Vector Mnchine (SVM) Dengan Penambahan Algoritma Levenshtein Distance”. Di dalam penelitian tersebut Manuaba menerangkan bahwa waktu proses pada tahapan *text preprocessing* dengan penambahan algoritma *Levenshtein Distance* lebihlama sebesar 295;606 detik, jika dibandingkan dengan tanpa menggunakan Algoritma *Levenshtein Distance*. Dengan nilai *accuracy* mengalami peningkatan sebesar 2%, nilai *recall* negatif 1%, nilai *recall* positif 3%, nilai *precision* negatif 2%, dan nilai *precision* positif 1%[11].

Dari beberapa penelitian sebelumnya mengenai topik yang sama yaitu membahas tentang algoritma *knuth morris pratt (KMP)* dan algoritma *levenshtein distance*, skenario dan sekaligus menjadikan penelitian sebelumnya sebagai tolak ukur sebagai capaiannya penulis untuk melakukan perbandingan kedua algoritma tersebut di dalam penelitian ini kedepanya.

## 2. METODE PENELITIAN

Pada bab ini penulis akan menjelaskan tentang analisis dan perancangan dari perbandingan algoritma *string matching knuth morris pratt* dan algoritma *levenshtein* pada aplikasi pengarsipan dan pencarian data anggota honda megapro club Indonesia Metode yang akan dipakai yaitu metode eksperimen yang terdiri dari tiga tahapan yaitu : (1) Analisis Kebutuhan, (2) Desain Sistem, (3) Penulisan Kode Program, (4) Pengujian Program, (5) Skema dan Perbandingan.



Gambar 2.1. Alur penelitian

Berdasarkan gambar 2,1. Alur Penelitian dijelaskan dimulai dengan pentingnya peran dari analisis kebutuhan yaitu proses dimana dilakukannya pengumpulan informasi atau kebutuhan apasaja yang diperlukan untuk penerapan dan perbandingan algoritma *Knuth Morris Pratt dan Levenshtein*, setelah melakukan analisis kebutuhan barulah berlanjut untuk melakukan desain sistem merupakan proses dimana dilakukannya perancangan disain sistem atau juga perancangan program sesuai dari apa yang di dihasilkan oleh analisis kebutuhan, kemudian berlanjut untuk penulisan kode program dimana proses tersebut

merupakan dilakukannya pembangunan aplikasi berdasarkan rancangan yang telah di buat, jika aplikasi telah selesai di bangun masuklah pada tahapan pengujian program dimana proses ini yaitu memastikan bahwa program telah berjalan dengan baik, setelah program lolos dan tidak ada lagi eror yang muncul berljaut ke skema perbandingan, dimana hal ini dilakukan untuk menentukan algoritma mana yang kecepatannya lebih unggul dalam melakukan pencarian data.

## 2.1. Analisis Kebutuhan

### 2.1.1. Pengumpulan Data

Dalam penelitian ini penulis mengumpulkan data yang dibutuhkan sebagai berikut :

1. Studi kepustakaan (*Literature*). Yaitu data yang diperoleh melalui *literature*, melakukan studi kepustakaan dalam mencari bahan bacaan dari buku, jurnal dan internet yang berkaitan sesuai dengan penelitian sebelumnya dan parameter yang akan diteliti.
2. Studi Dokumen Sekunder, Yaitu Penulis mendapatkan data penelitian melalui Youtube dan Divisi Anggota Pusat Honda Megapro Club Indonesia (HMPCI).
3. Uji Coba (*Testing*). Data-data Hasil Perbandingan *String Macthing* antara *Knuth Morris Pratt* dengan *Levenshtein Distance*

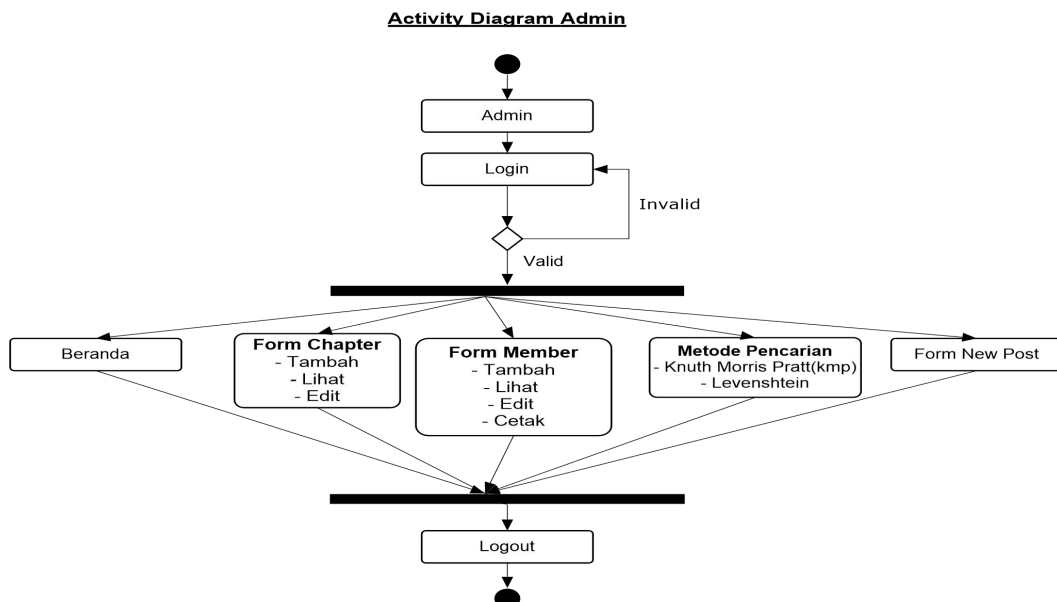
### 2.1.2. Pengolahan Data Awal

Data yang diperoleh dari Divisi Anggota Honda Megapro Club Indonesia yaitu data sebagian anggota dari awal terbentuk komunitas tersebut kurang lebih data yang di terima sebanyak 1.500 dengan atribut Id Anggota, Nama, dan Asal Chapter

## 2.2. Desain Sistem

### 2.2.1. Rancangan Aplikasi

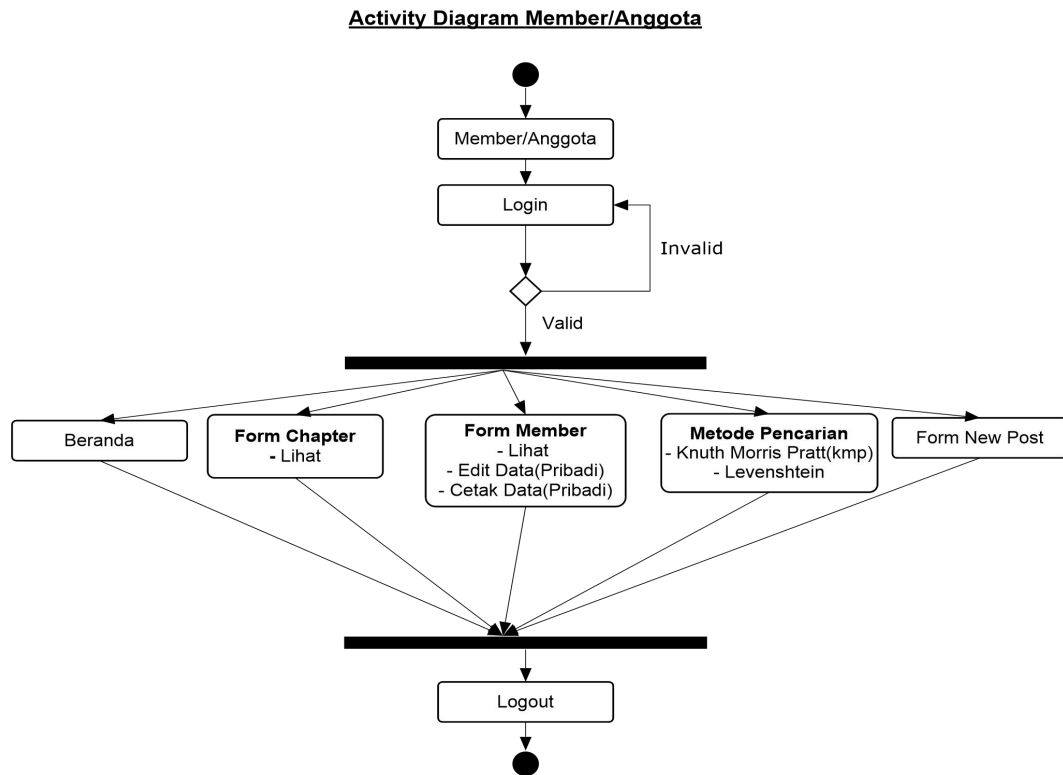
Rancangan aplikasi yang di rancang untuk membedakan akses yang dimiliki admin dan user(anggota/member) yang akan di gambarkan dalam bentuk *activity diagram* :



Gambar 2.2. Activity Diagram Admin

Pada gambar diatas merupakan activity diagram yang menjejelaskan apa saja yang dapat di akses oleh admin. Dibagian ini admin dapat mengakses semua fitur

yang berada di aplikasi tersebut diantaranya dapat menambahkan, mengedit, melihat ataupun mencetak data tersebut.



Gambar 2.3. Activity Diagram Member/Anggota

Pada gambar activity diagram diatas merupakan gambar yang menjelaskan tentang apa saja yang dapat di akses oleh anggota/member. Dibagian ini anggota/member dapat mengakses seluruh form di dalam aplikasi termasuk melakukan pencarian data namun hanya dapat membuat postingan/artikel, melihat, mengedit dan mencetak data sendiri. tidak dapat menambahkan data anggota ataupun daftar chapter.

### 2.3. Penulisan Kode Program

Pada tahapan ini merupakan tahap penerjemahan desain sistem yang telah dibuat kedalam bentuk kode perintah-perintah yang dimengerti komputer dengan menggunakan bahasa pemrograman

### 2.4. Pengujian Program

Pada tahapan ini yaitu dilakukannya pengujian aplikasi tersebut untuk memastikan bahwa aplikasi yang dibuat telah sesuai dengan desain dan semua fungsi dapat digunakan dengan yang di harapkan.

### 2.5. Skema Perbandingan

Skema perbandingan yang di lakukan untuk membandingkan algoritma *knuth morris pratt* dan algoritma *levenshtein* yaitu, dengan menguji kecepatan waktu kerja kedua algoritma dalam menemukan data anggota honda megapro club indonesia yang di cari.

### 3. HASIL DAN PEMBAHASAN

Dalam bab ini membahas mengenai tahapan hasil dan pembahasan dari perbandingan algoritma string matching knuth morris pratt dan algoritma levenshtein pada aplikasi pengarsipan dan pencarian data honda megapro club Indonesia.

#### 3.1. Hasil Penelitian

##### 3.1.1. Hasil Desain Aplikasi

##### 1. Hasil Desain Form Login

Gambar 3.1. Hasil Tampilan Form Login.

##### 2. Hasil Desain Form Dashboard

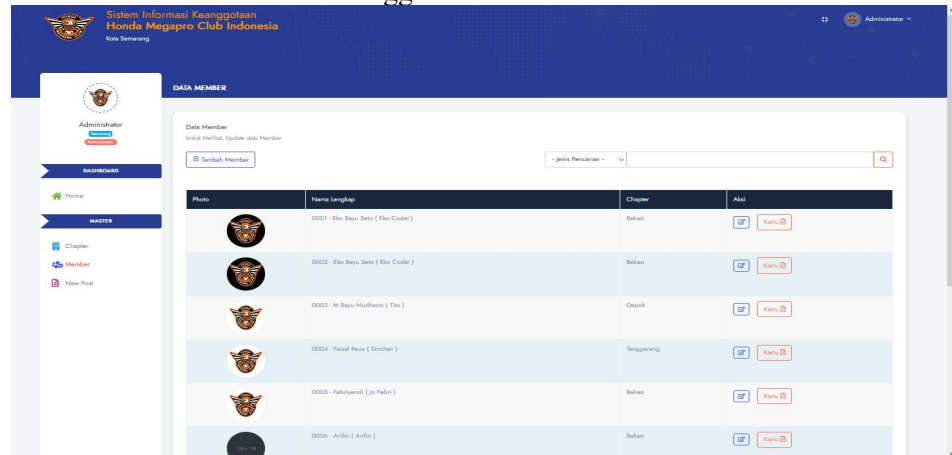
Gambar 3.2. Hasil Tampilan Form Menu Utama (Dashboard).

##### 3. Hasil Desain Form Chapter

Kode Chapter	Nama Chapter	Aksi
0001	Jakarta	<a href="#">Lihat Data</a>
0002	Bandung	<a href="#">Lihat Data</a>
0003	Semarang	<a href="#">Lihat Data</a>
0004	Sidoarjo	<a href="#">Lihat Data</a>
0005	Bekasi	<a href="#">Lihat Data</a>

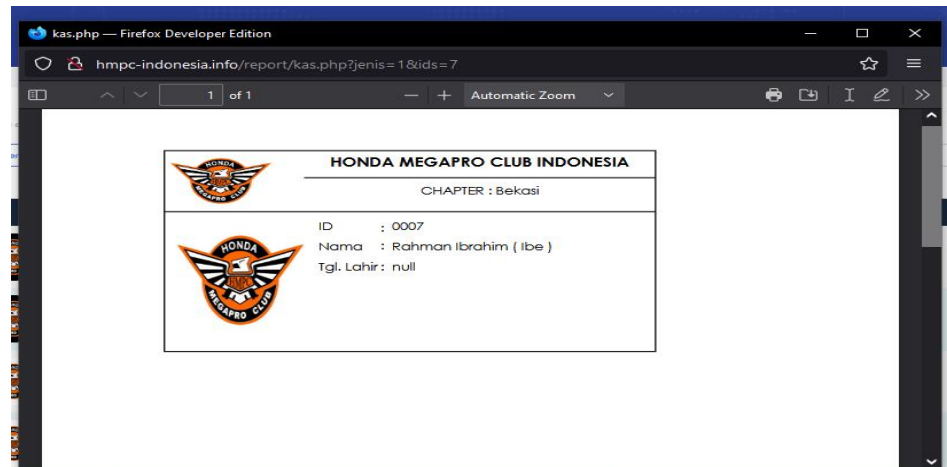
Gambar 3.3. Hasil Tampilan Form Chapter.

4. Hasil Desain Form Member/Anggota



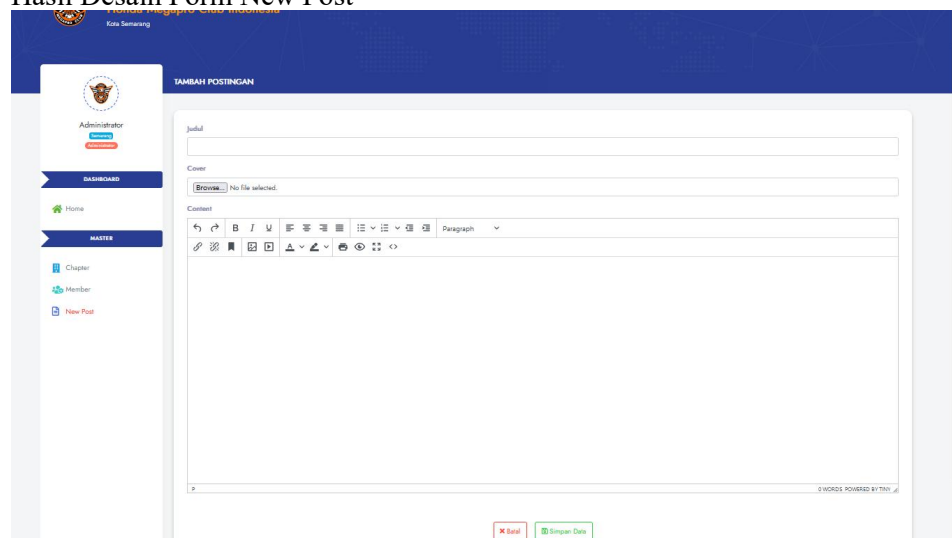
Gambar 3.4. Hasil Tampilan Form Member.

5. Hasil Desain Form Cetak Kartu



Gambar 3.5. Hasil Tampilan Form Cetak Kartu.

6. Hasil Desain Form New Post



Gambar 3.6. Hasil Tampilan Form New Post.

### 3.2. Pembahasan

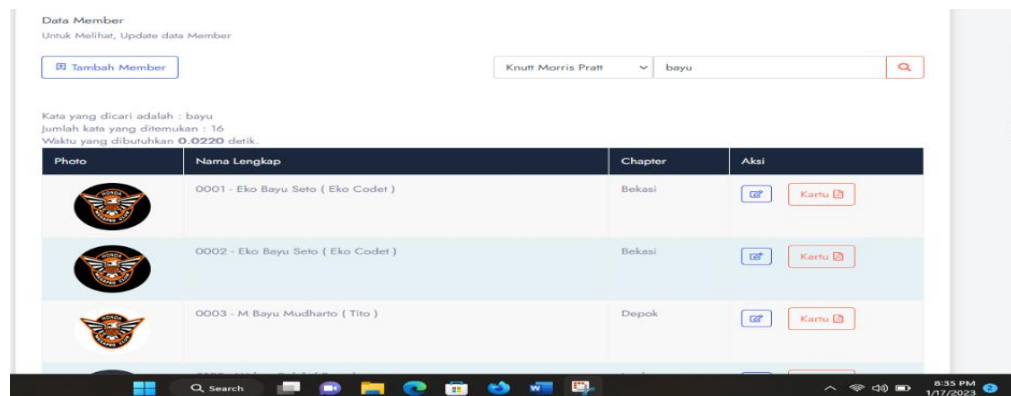
Pembahasan dalam penelitian ini lebih berfokus pada perbandingan peforma algoritma *knuth morris pratt* dan algoritma levenshtein. Pada pengujian ini dilakukan dengan membandingkan waktu kecepatan dan ketepatan dalam menemukan data di antara 1.500 data yang di cari. Dari penjelasan tersebut didapatkan hasil perbandingan kerja kedua algoritma yaitu.

#### 1. Algoritma Knuth Morris Pratt

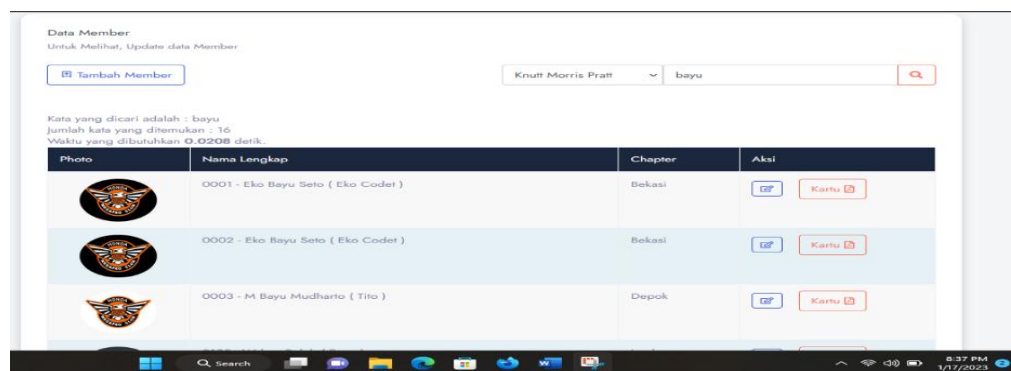
Dari hasil pengujian pencarian data menggunakan algoritma *Knuth Morris Pratt* dengan kata yang sama dalam di antara 1.500 data anggota didapatkan hasil kecepatan 0,0220 detik, 0.0208 detik, 0.0219 detik, 0,0220 detik, 0,0219 detik

Tabel 3.1. Hasil pencarian data menggunakan algoritma *knuth morris pratt* (KMP)

Pengujian Ke -	Lama Waktu Pencarian Data Menggunakan Algoritma Knuth Morris Pratt (KMP)	Status
1	0,0220 Detik	Ditemukan
2	0,0208 Detik	Ditemukan
3	0,0219 Detik	Ditemukan
4	0,0220 Detik	Ditemukan
5	0,0219 Detik	Ditemukan

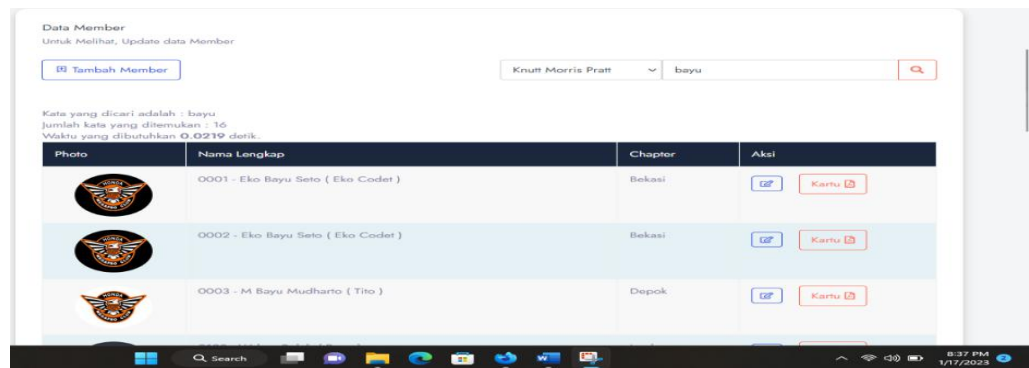


Gambar 3.9. Hasil Pencarian ke-1 algoritma *Knuth Morris Pratt*

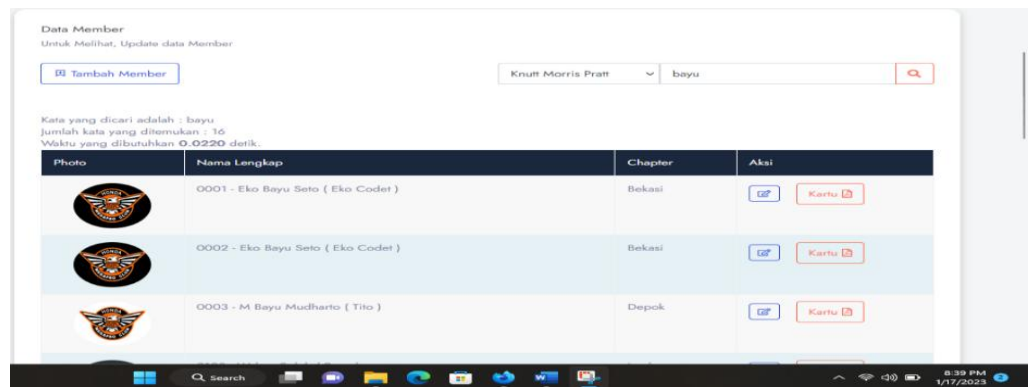


Gambar 3.10. Hasil Pencarian ke-2 algoritma *Knuth Morris Pratt*

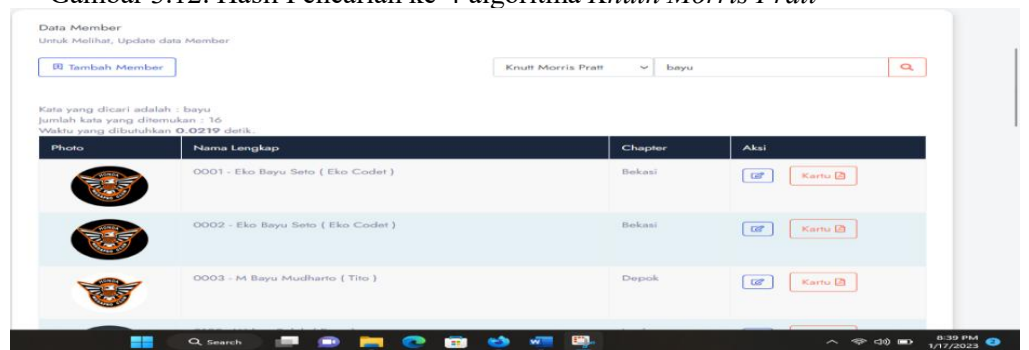




Gambar 3.11. Hasil Pencarian ke-3 algoritma *Knuth Morris Pratt*



Gambar 3.12. Hasil Pencarian ke-4 algoritma *Knuth Morris Pratt*



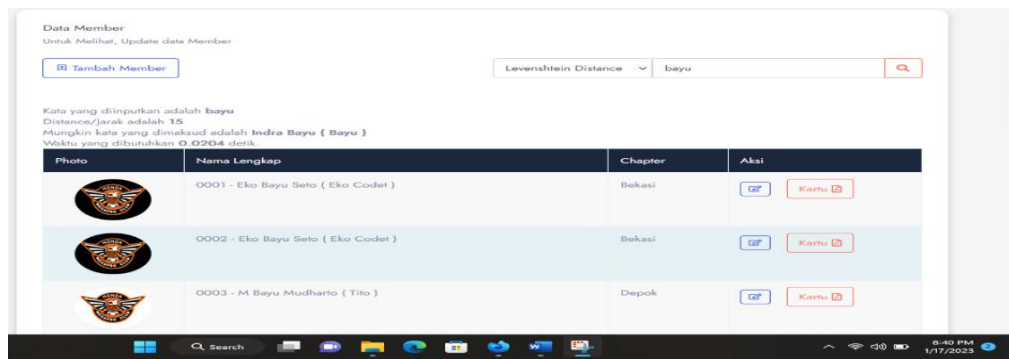
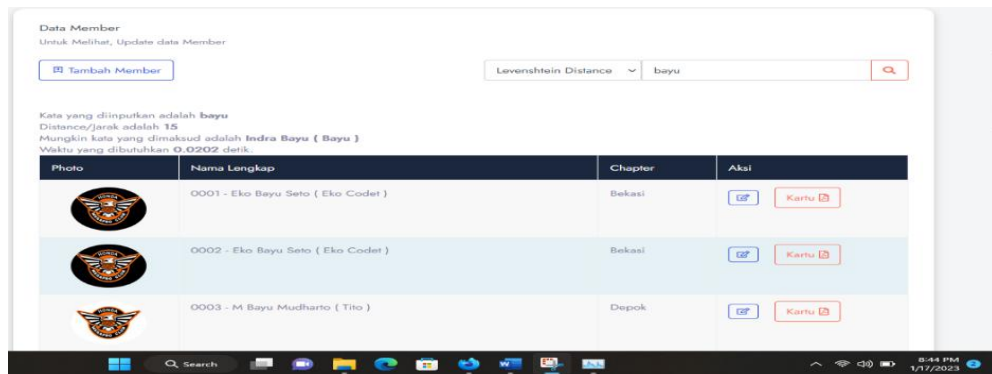
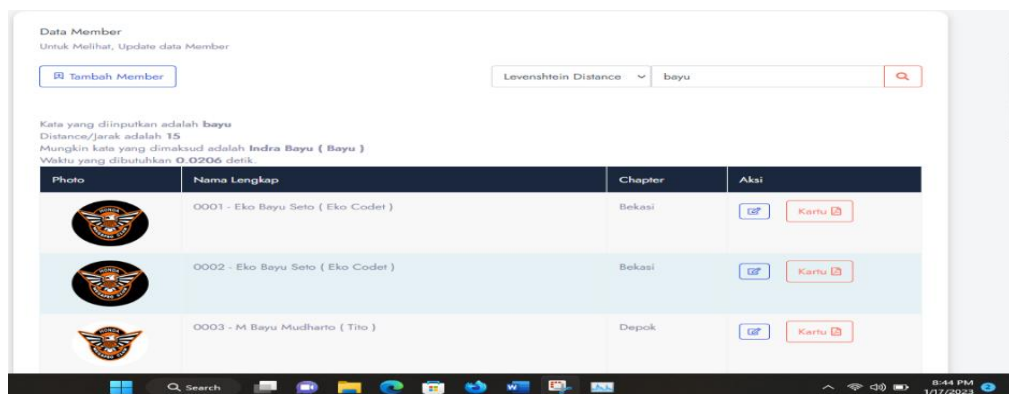
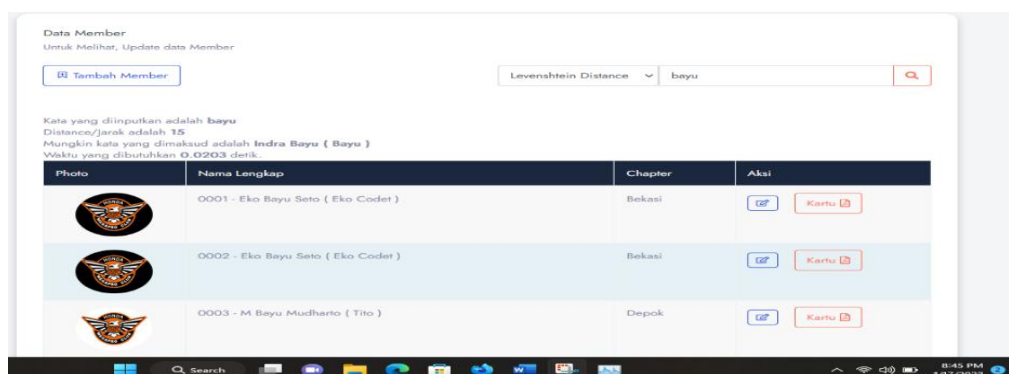
Gambar 3.13. Hasil Pencarian ke-5 algoritma *Knuth Morris Pratt*

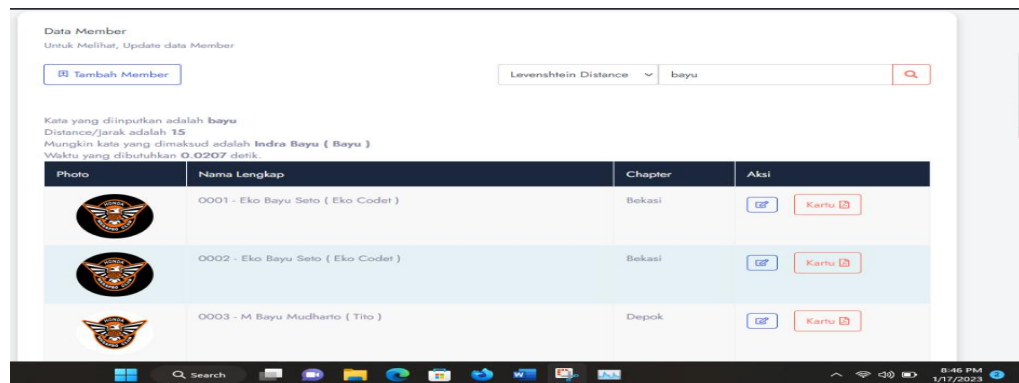
2. Algoritma Levenshtein

Dari Hasil pengujian pencarian data menggunakan algoritma levenshtein di dapatkan hasil kecepatan dalam proses pencarian data yaitu 0.0204 detik, 0.0202 detik, 0.0206 detik, 0,0203 detik, 0,0207 detik.

Tabel 3.2. Hasil pencarian data menggunakan algoritma *levenshtein*

Pengujian Ke -	Lama Waktu Pencarian Data Menggunakan Algoritma <i>Levenshtein</i>	Status
1	0,0204 Detik	Ditemukan
2	0,0202 Detik	Ditemukan
3	0,0206 Detik	Ditemukan
4	0,0203 Detik	Ditemukan
5	0,0207 Detik	Ditemukan

Gambar 3.14. Hasil Pencarian ke-1 algoritma *Levenshtein*Gambar 3.15. Hasil Pencarian ke-2 algoritma *Levenshtein*Gambar 3.16. Hasil Pencarian ke-3 algoritma *Levenshtein*Gambar 3.17. Hasil Pencarian ke-4 algoritma *Levenshtein*



Gambar 3.18. Hasil Pencarian ke-5 algoritma *Levenshtein*

3. Hasil Perbandingan

Table 3.3 Hasil Perbandingan Algoritma

Pengujian Ke -	Lama Waktu Pencarian Data	
	Algoritma <i>Knuth Morris Pratt</i>	Algoritma <i>Levenshtein</i>
1	0,0220 Detik	0,0204 Detik
2	0,0208 Detik	0,0202 Detik
3	0,0219 Detik	0,0206 Detik
4	0,0220 Detik	0,0203 Detik
5	0,0219 Detik	0,0207 Detik

4. KESIMPULAN DAN SARAN

4.1. Kesimpulan

Dari hasil uji komparasi algoritma *Knuth Morris Pratt* (KMP) dan *Levenshtein Distance*, terlihat bahwa Algoritma *Levenshtein Distance* lebih direkomendasikan daripada algoritma KMP dalam hal kecepatan dan akurasi pencarian data member. Bahkan jika perbedaan waktu tidak terlalu besar, *Levenshtein Distance* memeriksa semua string/kata di target, menghitung jarak antara sumber dan target, lalu menambahkan karakter, menghapus karakter, atau bahkan mengganti karakter. Anda dapat menggunakan algoritma *Levenshtein* untuk meminimalkan kesalahan saat mencari data.

4.2. Saran

Berdasarkan hasil pengujian dan analisa, peneliti mengusulkan beberapa saran yang sekiranya bermanfaat untuk peneliti selanjutnya :

1. Algoritma *Knuth Morris Pratt* (KMP) memang sudah terbilang cepat saat melakukan pencarian data dalam sebuah kalimat namun tidak ada kata petunjuk kalimat apa yang berada dalam kalimat tersebut, jadi mungkin boleh di modifikasi lagi agar memberikan petunjuk kepada pengguna jikalau sumber yang dicari mempunyai kemiripan dengan target yang dicari seperti yang dilakukan Algoritma *Levenshtein Distance*.
2. Algoritma *Levenshtein Distance* sangat unggul dalam memproteksi plagiat karena seluruh kalimat dihitung jaraknya antara sumber dan target, namun kekurangannya sedikit lebih lama karena melakukan Penghitungan, penambahan, penghapusan dan penggantian karakter ke dalam target, sehingga algoritma ini

masih bisa modifikasi atau dikembangkan agar dapat memproses kalimat lebih cepat lagi.

#### UCAPAN TERIMA KASIH

Puji syukur penulis panjatkan kehadirat Allah SWT atas kekuatan jasmani dan rohani serta anugrah yang melimpah. Tak lupa saya ucapkan terima kasih kepada kedua orang tua saya atas dukungan dan semangatnya kepada penulis. Penulis juga mengucapkan terima kasih kepada Ibu Nurul Huda selaku pembimbing dan staf serta manajemen Universitas Bina Darma Palembang untuk berpartisipasi dalam proyek penelitian ini sehingga berjalan dengan lancar. Juga, terima kasih kepada diri saya sendiri karena telah bertahan sampai sekarang.

#### DAFTAR PUSTAKA

- [1] “Honda MegaPro Club (HMPC) Indonesia.” <https://www.motormobile.net/more.php?id=13054> (accessed Mar. 16, 2023).
- [2] K. Telaumbanua, F. N. Damanik, M. Ilhami, and E. Sufarnap, “Penerapan Algoritma Knuth-Morris-Pratt Dalam Sistem Jasa Konsultan berbasis Mobile dan Web,” vol. 4, no. 2, 2023.
- [3] A. Rofiqih, M. Z. Siambaton, and T. Haramaini, “Penerapan Algoritma Levenshtein String pada E-Arsip Kecamatan Pagar Merbau,” *sudo j. Teknik inform.*, vol. 1, no. 1, pp. 1–7, Mar. 2022, doi: 10.56211/sudo.v1i1.1.
- [4] M. Ilham and A. H. Mirza, “Penerapan Algoritma Knuth Morris Pratt Dalam Fitur Pencarian Pengarsipan Dokumen Pada Sma Plus Negeri 17 Palembang,” *Journal-SEA*, vol. 1, no. 2, pp. 110–121, May 2020, doi: 10.51519/journalsea.v1i2.49.
- [5] N. K. Sumiari and N. K. D. Ari Jayanti, “Optimasi Dashboard Information System Stikom Bali Dengan Algoritma Levenshtein Distance,” *citec*, vol. 6, no. 1, p. 12, Apr. 2020, doi: 10.24076/citec.2019v6i1.181.
- [6] A. A. Nababan and M. Jannah, “ALGORITMA STRING MATCHING BRUTE FORCE DAN KNUTH-MORRIS-PRATT SEBAGAI SEARCH ENGINE BERBASIS WEB PADA KAMUS ISTILAH JARINGAN KOMPUTER,” *Jurnal Mantik Penusa*, vol. 3, pp. 87–94, 2019.
- [7] A. Qurania, T. -, and E. Candra, “Kamus Digital Tanaman Obat Menggunakan Algoritme Knuth Morris Pratt Berbasis Mobile,” *JIIKM*, vol. 19, no. 1, pp. 54–64, Jul. 2022, doi: 10.33751/komputasi.v19i1.4269.
- [8] D. S. Islamiyati and A. Fikri, “Penerapan Algoritma Knuth-Morris-Pratt dalam Mendeteksi Tingkat Kemiripan Judul Skripsi Berbasis Web,” *josh*, vol. 3, no. 2, pp. 58–63, Jan. 2022, doi: 10.47065/josh.v3i2.1168.
- [9] C. Natalie, V. C. Mawardi, and M. D. L. Sitorus, “OPTICAL CHARACTER RECOGNITION MENGGUNAKAN UIPATH DAN PENCOCOKAN DATA SERTIFIKAT DENGAN ALGORITMA LEVENSHTTEIN DISTANCE,” vol. 1, no. 1, 2023, doi: <https://doi.org/10.24912/jsstk.v1i1.22747>.
- [10] W. Kuswanto, “IMPLEMENTASI ALGORITMA LEVENSHTTEIN DISTANCE DENGAN RESTFUL WEB SERVICE PADA KATA BAHASA INDONESIA KE BAHASA JAWA BERBASIS WEB,” *j.sist.teknol.inf.*, vol. 5, no. 2, pp. 78–85, Aug. 2020, doi: 10.32528/justindo.v5i2.3447.
- [11] I. B. N. W. Manuaba, G. R. Dantes, and G. Indrawan, “Analisis Sentimen Data Provider Layanan Internet Pada Twitter Menggunakan Support Vector Machine Dengan Penambahan Algoritma Levenshtein Distance,” *siskom-kb*, vol. 5, no. 2, pp. 9–17, Mar. 2022, doi: 10.47970/siskom-kb.v5i2.261.