

## **String Matching Pada Pencarian Di Web Jurnal Menggunakan Algoritma Knuth Morris Pratt (KMP)**

Rizky Dharma Sanjaya<sup>1</sup>; Ratih Titi Komala Sari<sup>1\*)</sup>

1. Informatika, Fakultas Teknologi Komunikasi dan Informasi, Universitas Nasional,  
Jl. Sawo Manila No.61, RT.14/RW.7, Pejaten Barat, Ps. Minggu, Jakarta Selatan,  
DKI Jakarta 12520, Indonesia

<sup>\*)</sup>Email: [ratih.titi@civitas.unas.ac.id](mailto:ratih.titi@civitas.unas.ac.id)

Received: 04 Juli 2023 | Accepted: 27 Agustus 2023 | Published: 17 November 2023

### **ABSTRACT**

*This research aims to: (1) enhance the responsiveness of a journal website for easier access from mobile devices and improve the quality of its appearance and user experience during web browsing, (2) facilitate easy access to information and visibility, and (3) improve journal data processing effectively by utilizing the Knuth Morris Pratt (KMP) algorithm. The subjects of this research are the user interface and user experience of the web application for students, learners, and the general public, while the object of this research is the web application itself. The data sources used consist of secondary data obtained from books, documents, and archives containing relevant information. This research employs data collection techniques by gathering information from pre-existing data sources such as books, journals, reports, the internet, and other documents. The estimated time required to conduct this research is two months (November-December 2022). This research develops a responsive web using the Laravel Framework and the Knuth Morris Pratt (KMP) algorithm in searching for journal data. The implementation of the KMP algorithm resulted in a search time of approximately 0.505 seconds with 15 keywords.*

**Keywords:** Knuth Morris Pratt, Responsive, Search Algorithm, String Matching, Journal website

### **ABSTRAK**

*Penelitian ini bertujuan untuk: (1) meningkatkan responsivitas website jurnal agar lebih mudah diakses dari perangkat mobile dan meningkatkan kualitas tampilan serta pengalaman pengguna saat menjelajah web, (2) memudahkan akses informasi dan visibilitas, dan (3) meningkatkan pengolahan data jurnal dengan baik memanfaatkan tata cara algoritma Knuth Morris Pratt (KMP). Subjek penelitian ini adalah user interface dan user experience pada aplikasi web jurnal untuk mahasiswa, pelajar, dan masyarakat umum, sedangkan objek penelitian ini adalah aplikasi web jurnal. Sumber data yang digunakan berupa data sekunder yang didapat dari buku, dokumen, dan arsip yang mengandung informasi relevan. Penelitian ini menggunakan teknik pengumpulan data dengan mengumpulkan informasi dari sumber data yang telah ada sebelumnya seperti buku, jurnal, laporan, internet, dan dokumen-dokumen lainnya. Waktu yang dibutuhkan untuk melakukan penelitian ini adalah dua bulan (November-Desember 2022). Penelitian ini mengembangkan web jurnal responsif menggunakan Framework Laravel dan algoritma Knuth Morris Pratt (KMP) dalam pencarian data jurnal. Implementasi algoritma KMP menghasilkan waktu pencarian sekitar 0,505 detik dengan 15 kata kunci.*

**Kata kunci:** Algoritma Pencarian, Knuth Morris Pratt, Pencocokan String, Responsif, Web Jurnal

## 1. PENDAHULUAN

Selama beberapa dekade terakhir, perkembangan teknologi informasi dan komunikasi telah mengubah dunia secara drastis menjadi masyarakat informasi [1]. Teknologi informasi adalah konsep integrasi Perkembangan teknologi informasi dan komunikasi mencakup segala aspek yang terkait dengan perangkat keras, perangkat lunak, dan layanan terkait. Dengan menggunakan peralatan tersebut, informasi digital dapat disimpan, dihitung, diubah, dan dipertukarkan [2].

Perkembangan teknologi informasi dan komunikasi memberikan dampak besar pada bermacam aspek kehidupan, seperti di bidang pendidikan dan ilmu pengetahuan. Berbagai hasil penelitian diterbitkan dalam wujud publikasi yang mudah diakses oleh masyarakat luas. Salah satu media yang digunakan untuk menerbitkan publikasi yaitu web jurnal. Media web jurnal digunakan sebagai sarana untuk mempublikasikan artikel dari berbagai penelitian. Web jurnal menerbitkan artikel melalui proses penelaahan sejawat atau *peer-review*, di mana artikel tersebut diperiksa dan direvisi oleh rekan sejawatnya untuk memastikan kualitas yang terjamin [3].

Adanya fitur pencarian yang mudah pada suatu web jurnal memungkinkan pembaca untuk menemukan artikel-artikel yang relevan dengan minat dan topik penelitiannya. Web jurnal yang responsif sangatlah dibutuhkan untuk memudahkan akses publikasi bagi peneliti yang banyak mengakses internet, terutama melalui perangkat *mobile*. Perangkat *mobile* memiliki kelebihan dalam hal portabilitas, kemudahan akses, dan fleksibilitas. Selain itu, perangkat *mobile* memiliki antarmuka yang intuitif sehingga pengguna tidak perlu memiliki keahlian khusus untuk mengoperasikannya. Perihal ini juga sejalan dengan tren kenaikan pemakaian perangkat *mobile* yang terus menggeser pemakaian *desktop*. Penggunaan perangkat *mobile* telah mencapai 62,1% di Indonesia, sedangkan penggunaan *desktop* hanya 37,5% [4]. Namun, beberapa web jurnal ditemukan kurang responsif serta sulit diakses melalui perangkat *mobile*.

Permasalahan tersebut menjadi dasar penelitian ini untuk merancang serta meningkatkan web jurnal berbasis *website* responsif memanfaatkan *Framework* dari Laravel. *Framework* Laravel adalah kerangka kerja PHP yang modular, aman, dan cepat. Dengan fitur dukungan akses *database* yang beragam, utilitas pengembangan aplikasi web, dan mesin routing yang sederhana, *Framework* Laravel memungkinkan pengembangan aplikasi yang skalabel dan efisien [5]. Struktur *Model-View-Controller (MVC)* pada *Framework* Laravel terdapat fitur *routing* yang menghubungkan permintaan dari pengguna dengan kontroler, sehingga kontroler tidak langsung menerima permintaan tersebut [6]. Pemanfaatan *Framework* Laravel dalam pengembangan web jurnal memberikan beberapa keuntungan diantaranya memudahkan pengelolaan *database* serta pemrograman yang lebih efektif. Tipe ini juga membantu pengembang untuk mengakses fitur-fitur yang lebih mutakhir semacam autentikasi pengguna, manajemen hak akses, dan lain sebagainya.

Penelitian ini didukung dengan penggunaan algoritma pencarian *string matching* untuk memudahkan pengguna dalam mencari informasi. Algoritma pencocokan *string (string matching)* adalah metode untuk menentukan kecocokan atau hasil dari satu ataupun lebih pola teks tertentu [7]. Algoritma *Knuth Morris Pratt (KMP)* adalah algoritma *string matching* yang melakukan pencarian pola kata dari awal hingga akhir dalam teks dengan cara mencocokkan kata yang bergerak dari kiri ke kanan. Teknik ini akan membandingkan pola kata yang dicari dengan teks secara berurutan dari awal teks dan akan melakukan pergeseran pola kata hingga pola kata tersebut ditemukan pada akhir teks [8]. Selama penelitian berlangsung, peneliti menemukan bahwa dengan menerapkan algoritma KMP, sistem dapat menampilkan nama-nama tumbuhan dalam bahasa Latin berdasarkan kata kunci (*keyword*) yang dimasukkan oleh *system-user* [9]. Fitur pencarian menggunakan algoritma KMP juga telah dikembangkan dalam sistem yang dibangun, dengan melakukan uji coba sebanyak 20 jenis

kelinci berbeda. Hasilnya, jenis algoritma ini mampu menyelesaikan pencarian dengan baik sehingga lebih efisien[10].

Sistem ini akan membantu meningkatkan efektivitas penggunaan algoritma *Knuth Morris Pratt*, yang pada akhirnya akan meningkatkan kinerja secara keseluruhan menjadi lebih baik.

## 2. METODE/PERANCANGAN PENELITIAN

### 2.1. Algoritma *Knuth Morris Pratt*

Algoritma *Knuth Morris Pratt* (KMP) ialah algoritma yang termasuk dalam jenis algoritma pencocokan *string*[11]. Algoritma KMP bekerja dengan mencocokkan karakter-karakter dalam pola dengan teks yang diberikan[12]. Dalam kata lain, algoritma ini membantu untuk mengetahui keberadaan suatu kata dalam teks dengan melakukan pengamatan awal (*preprocessing*) untuk memeriksa kata sebelumnya. Algoritma KMP melaksanakan pemeriksaan kata dari kiri ke kanan[13], dengan menggunakan tabel kecocokan yang telah dibangun berdasarkan pola yang hendak dicari. Berikut adalah langkah-langkah yang dilakukan oleh algoritma KMP saat melakukan pencocokan *string*[14]:

1. Memasukkan kata kunci yang ingin dicari (*Query*). Dalam hal ini, P adalah *pattern* atau rangkaian kata yang menjadi contoh atau *pattern* teks yang akan dicari, sedangkan T adalah *text* ataupun judul dokumen.
2. Algoritma KMP mulai membandingkan pola atau rangkaian kata yang diberikan pada bagian awal teks.
3. Dalam algoritma ini, pencocokan karakter dilakukan secara satu per satu dari kiri ke kanan antara pola dan karakter dalam teks yang sesuai, hingga salah satu keadaan berikut terpenuhi:
  - Jika terjadi perbedaan (*mismatch*) antara karakter dalam *pattern* dan karakter dalam teks yang sedang dibandingkan.
  - Semua karakter dalam pola cocok. Algoritma akan memberitahukan bahwa pencocokan ditemukan di posisi ini.
4. Setelah itu, algoritma akan memindahkan pola, dan langkah-langkah dari nomor 2 akan diulang hingga pola berada di ujung teks.

Metode penelitian algoritma ini memungkinkan pencarian pola yang efisien dengan kompleksitas waktu  $O(n + m)$ , di mana  $n$  yakni panjang teks dan  $m$  merupakan panjang pola[15].

## 3. HASIL DAN PEMBAHASAN

### 3.1. Implementasi Algoritma *String Matching Knuth Morris Pratt*

Pada bagian ini, akan diilustrasikan pemanggilan algoritma *string matching* dan pengimplementasian algoritma KMP dalam *pseudocode* program untuk mencari pola atau *substring* dalam sebuah teks. Berikut adalah *pseudocode* algoritma *string matching* berdasarkan kode yang telah dibuat.

```
function search($request)
    keyword = request('keyword')
    IF keyword ada, THEN
        jurnal = Jurnal::where('judul', 'LIKE', .keyword.)
        ->OR ('ket', 'LIKE', .keyword.)
        ->OR ('tahun', 'LIKE', .keyword.)
        ->OR (function (query) gunakan keyword
```

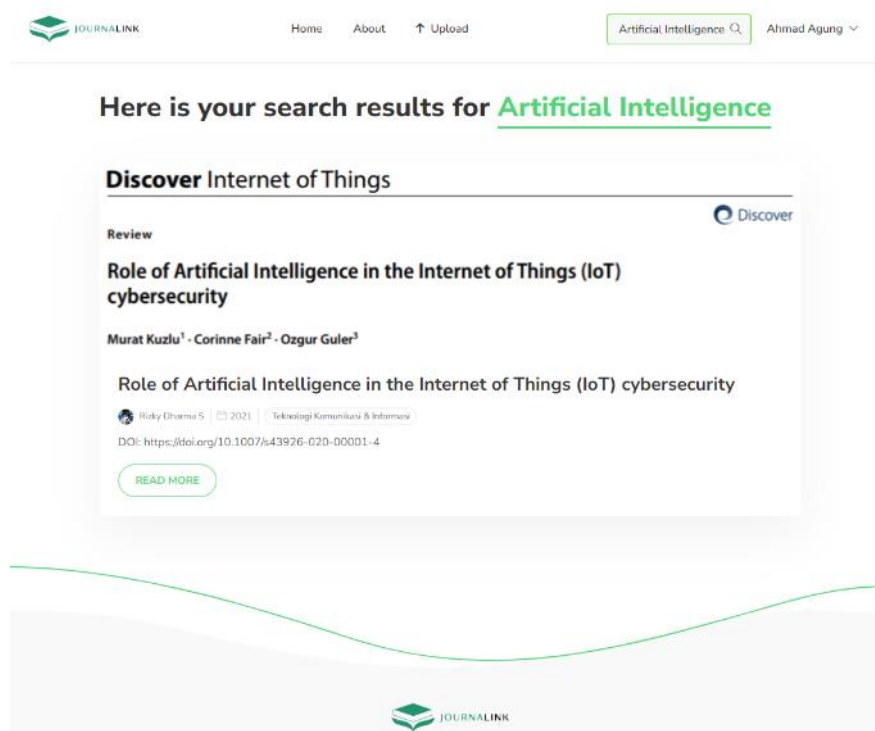
```

        query->whereHas('profile', function (query2) gunakan keyword
            query2->where('nama', 'LIKE', .keyword.)
        )
    )
    ->paginate(10)
ELSE
    jurnal = Jurnal::paginate(10)
END IF

```

RETURN tampilan ke ('pages.searchResult') dengan data jurnal dan keyword, serta judul "Search Jurnal"

END function



Gambar 1. Output yang diperoleh menggunakan algoritma string matching

Pada Gambar 1 ditampilkan hasil *output* dari *pseudocode* proses pencarian menggunakan algoritma. *Output* ini diperoleh dari algoritma *string matching* pada fungsi yang menerima permintaan sebagai input dan mencari jurnal berdasarkan kata kunci (*keyword*) yang diberikan. Jika kata kunci ditemukan, maka dilakukan pencarian dengan menggunakan operasi “LIKE” pada beberapa atribut jurnal (judul, keterangan, tahun), serta menggunakan relasi dengan profil untuk mencari berdasarkan nama. Hasil pencarian kemudian disajikan dalam bentuk halaman yang dipaginasi. Jika kata kunci tidak ditemukan, maka semua jurnal akan ditampilkan dalam halaman yang dipaginasi. Pada hasilnya, pencarian ditampilkan pada halaman “pages.searchResult” yang mencakup data jurnal, kata kunci, dan judul halaman. Berikut adalah *pseudocode* pengimplementasian algoritma *string matching* KMP.

```

CLASS KMPController
function kmpSearch(text, pattern)

```

```

n = PanjangTeks(text)
m = PanjangPola(pattern)
lps = computeLPSArray(pattern)
i = 0 // Indeks untuk teks[]
j = 0 // Indeks untuk pola[]
positions = [] // Array untuk menyimpan posisi kemunculan pola dalam teks

WHILE i < n
    IF pola[j] sama dengan teks[i], THEN
        i++
        j++
    END IF

    IF j sama dengan m THEN
        // Pola ditemukan di indeks i - j
        positions.tambahkan(i - j),
        j = lsp[j - 1]
    ELSE IF
        IF i < n dan pola[j] tidak sama dengan teks[i], THEN
            IF j bukan 0, THEN
                j = lsp[j - 1]
            ELSE
                i++
            END IF
        END IF
    END IF
END WHILE

RETURN positions
END function

```

```

function computeLPSArray(pattern)
    len = 0 // Panjang awalan dan akhiran terpanjang sebelumnya
    m = PanjangPola(pattern)
    lsp = []
    lsp[0] = 0
    i = 1
    WHILE i < m
        IF pola[i] sama dengan pola[len], THEN
            len++
            lsp[i] = len
            i++
        ELSE
            Jika len bukan 0, THEN
                len = lsp[len - 1]
            ELSE
                lsp[i] = 0, THEN

```

```

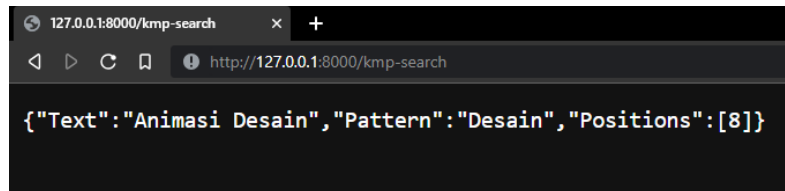
        i++
    END IF
END IF
END WHILE

RETURN lps
END FUNCTION

Fungsi search()
    text = "Animasi Desain"
    pattern = "Desain"
    positions = kmpSearch(text, pattern)

    RETURN respons JSON dengan text, pattern, dan positions
END FUNCTION
END CLASS

```



**Gambar 2.** Output yang diperoleh menggunakan algoritma Knuth Morris Pratt

| Hasil Iterasi Algoritma Knuth Morris Pratt |   |   |   |   |   |   |   |   |   |   |   |   |
|--------------------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|
| Iterasi ke-1                               |   |   |   |   |   |   |   |   |   |   |   |   |
| a                                          | n | i | m | a | s | i | d | e | s | a | i | n |
| d                                          | e | s | a | i | n |   |   |   |   |   |   |   |
| Iterasi ke-2                               |   |   |   |   |   |   |   |   |   |   |   |   |
| a                                          | n | i | m | a | s | i | d | e | s | a | i | n |
|                                            | d | e | s | a | i | n |   |   |   |   |   |   |
| Iterasi ke-3                               |   |   |   |   |   |   |   |   |   |   |   |   |
| a                                          | n | i | m | a | s | i | d | e | s | a | i | n |
|                                            |   | d | e | s | a | i | n |   |   |   |   |   |
| Iterasi ke-4                               |   |   |   |   |   |   |   |   |   |   |   |   |
| a                                          | n | i | m | a | s | i | d | e | s | a | i | n |
|                                            |   |   | d | e | s | a | i | n |   |   |   |   |
| Iterasi ke-5                               |   |   |   |   |   |   |   |   |   |   |   |   |
| a                                          | n | i | m | a | s | i | d | e | s | a | i | n |
|                                            |   |   |   | d | e | s | a | i | n |   |   |   |
| Iterasi ke-6                               |   |   |   |   |   |   |   |   |   |   |   |   |
| a                                          | n | i | m | a | s | i | d | e | s | a | i | n |
|                                            |   |   |   |   | d | e | s | a | i | n |   |   |
| Iterasi ke-7                               |   |   |   |   |   |   |   |   |   |   |   |   |
| a                                          | n | i | m | a | s | i | d | e | s | a | i | n |
|                                            |   |   |   |   |   | d | e | s | a | i | n |   |
| Iterasi ke-8                               |   |   |   |   |   |   |   |   |   |   |   |   |
| a                                          | n | i | m | a | s | i | d | e | s | a | i | n |
|                                            |   |   |   |   |   |   | d | e | s | a | i | n |
| Run Time: 0,464 s                          |   |   |   |   |   |   |   |   |   |   |   |   |

**Gambar 3.** Hasil iterasi algoritma Knuth Morris Pratt

Gambar 2 menampilkan *output Pseudocode* representasi struktur dari kode *KMPController* yang mengimplementasikan algoritma KMP untuk mencari pola dalam teks.

Hasil iterasi algoritma KMP seperti yang ditunjukkan pada Gambar 3 untuk mencari pola “Desain” dalam teks “Animasi Desain” adalah sebagai berikut:

1. Pada setiap iterasi, algoritma KMP membandingkan karakter-karakter dalam pola “Desain” dengan karakter-karakter dalam teks “Animasi Desain”. Pencocokan dilakukan dari kiri ke kanan.
2. Iterasi ke-1 sampai ke-7: Tidak ada kecocokan antara karakter-karakter dalam pola “Desain” dengan karakter-karakter dalam teks “Animasi Desain”. Pola bergeser maju saat pencarian tidak cocok.
3. Iterasi ke-8: Terjadi kecocokan karakter pada indeks ke-8 dalam teks, yaitu “D” dengan karakter pertama dalam pola “Desain”. Kemudian dilanjutkan dengan pencocokan seluruh karakter pola “Desain” dengan karakter-karakter dalam teks. Hasilnya, pola “Desain” ditemukan pada indeks ke-8 dalam teks “Animasi Desain”.

Sehingga, pencarian pola “Desain” dalam teks “Animasi Desain” berhasil ditemukan pada iterasi ke-8.

### 3.2. Pengujian algoritma Knuth Morris Pratt

Pengujian algoritma *string matching* KMP pada aplikasi web jurnal dapat dilakukan dengan memasukkan informasi dari judul jurnal, tahun terbit, *author*, pengguna, nomor DOI, dan kata kunci yang ingin dicari ke dalam formulir pencarian. Setelah itu, sistem hendak menunjukkan hasil dari pencarian. Hasil pengujian tersebut ditampilkan dalam bentuk tabel 1.

**Tabel 1.** Pengujian kinerja algoritma *string matching* KMP pada fungsi data jurnal

| Pencarian ke-            | Kata kunci | Estimasi waktu (detik) |
|--------------------------|------------|------------------------|
| 1                        | aktivitas  | 0,492                  |
| 2                        | desain     | 0,464                  |
| 3                        | putri      | 0,437                  |
| 4                        | pengaruh   | 0,564                  |
| 5                        | berbasis   | 0,523                  |
| 6                        | pemahaman  | 0,538                  |
| 7                        | studi      | 0,415                  |
| 8                        | David      | 0,499                  |
| 9                        | DOI        | 0,441                  |
| 10                       | fakultas   | 0,575                  |
| 11                       | Ali        | 0,383                  |
| 12                       | arsitektur | 0,526                  |
| 13                       | Andika     | 0,478                  |
| 14                       | 2022       | 0,508                  |
| 15                       | a          | 0,725                  |
| Estimasi waktu rata-rata |            | 0,505                  |

Pada pengujian kinerja algoritma *string matching* KMP ini, dilakukan pencarian menggunakan 1018 data jurnal dan 15 kata kunci berbeda yang terdiri dari beberapa karakter. Hasil estimasi waktu menunjukkan bahwa algoritma KMP dapat melakukan pencarian data secara cepat. Pencarian menggunakan kata “Ali” membutuhkan waktu tercepat yaitu 0,383 detik, sedangkan kata “a” membutuhkan waktu terlama yaitu 0,725 detik. Hal ini disebabkan oleh banyaknya penggunaan kata

"a" dalam jurnal, author, dan nama pengguna. Rata-rata estimasi waktu pencarian data oleh algoritma KMP ialah 0,505 detik. Oleh karena itu, dapat disimpulkan bahwa algoritma pencocokan *string* KMP sesuai dan efisien dalam pencarian data jurnal dalam aplikasi web jurnal.

#### 4. KESIMPULAN DAN SARAN

Dari hasil penelitian ini, dapat diambil disimpulkan bahwa pengembangan web jurnal responsif menggunakan *Framework* Laravel dan penerapan algoritma KMP dalam pencarian informasi jurnal memberikan hasil yang signifikan. Penelitian ini berhasil mengatasi masalah responsivitas pada perangkat *mobile* serta meningkatkan kinerja pencarian data. Dalam sistem yang telah dibangun, fitur pencarian telah diimplementasikan dengan menggunakan algoritma untuk memberikan hasil pencarian yang cepat. Hasil implementasi algoritma KMP menunjukkan estimasi waktu rata-rata pencarian sekitar 0,505 detik dalam pengujian menggunakan 15 kata kunci sebagai acuan. Diharapkan pengembangan web jurnal responsif dengan penerapan algoritma KMP ini dapat memberikan kontribusi dalam meningkatkan kinerja dan pengalaman pengguna secara keseluruhan. Berikut beberapa rekomendasi untuk penelitian selanjutnya, yaitu:

1. Mengadopsi prinsip desain *mobile-first*, yaitu merancang web jurnal dengan memprioritaskan pengalaman pengguna pada perangkat *mobile* terlebih dahulu. Dengan memulai desain dari perangkat *mobile*, kemudian mengembangkannya untuk perangkat lain, akan memastikan bahwa web jurnal memiliki tampilan yang optimal pada semua perangkat.
2. Menghubungkan artikel dengan profil penulis di platform akademik untuk meningkatkan visibilitas.
3. Menyediakan opsi unggah artikel ke repositori institusi atau repositori umum untuk berbagi penelitian.

Dengan implementasi saran-saran ini, diharapkan penelitian selanjutnya dapat memperbaiki dan mengembangkan web jurnal responsif serta memanfaatkan algoritma KMP secara optimal dalam konteks pencarian data jurnal.

#### DAFTAR PUSTAKA

- [1] R. Bahrini dan A. A. Qaffas, "Impact of information and communication technology on economic growth: Evidence from developing countries" *Economies*, vol. 7, no. 1, Mar. 2019, doi: 10.3390/economies7010021.
- [2] Y. Hao, Y. Guo, dan H. Wu, "The role of information and communication technology on green total factor energy efficiency: Does environmental regulation work?" *Business Strategy and the Environment*, Wiley Blackwell, vol. 31(1), pp. 403-424, Jan. 2022, doi: 10.1002/bse.2901.
- [3] D. Setia Budi dan Ardiansyah, "Pengembangan Sistem Authoring Tools untuk Publikasi Jurnal Ilmiah Berbasis Web" *Jurnal Sarjana Teknik Informatika*, vol. 7, no. 2, pp. 38-48, Jun. 2019, doi: <http://dx.doi.org/10.12928/jstie.v7i2.15825>.
- [4] Statcounter.com. Maret 2023. *Desktop vs Mobile vs Tablet Market Share Indonesia*. Diakses pada 4 April 2023, dari <https://gs.statcounter.com/platform-market-share/desktop-mobile-tablet/indonesia>.
- [5] Z. Subecz, "Web-development with Laravel framework" *Gradus*, vol. 8, no. 1, pp. 211-218, 2021, doi: 10.47833/2021.1.csc.006.
- [6] A. L. Yudanto, H. Tolle, dan A. H. Brata, "Rancang Bangun Aplikasi Sistem Informasi Manajemen Laboratorium Biomedik Fakultas Kedokteran Universitas Brawijaya", *J-PTIHK*, vol. 1, no. 8, pp. 628-634, Jun. 2017. [Daring]. Tersedia pada: <http://j-ptiik.ub.ac.id>.
- [7] A. R. Hakim, K. Nasution, O. K. Sulaiman, dan M. Z. Siambaton, "Perancangan Aplikasi Skripsi Online Menggunakan Algoritma String Matching Knuth Morris-Pratt Pada Fakultas Teknik Universitas Islam

- Sumatera Utara” *ALGORITMA: Jurnal Ilmu Komputer dan Informatika*, vol. 3, no. 2, p. 98, Nov. 2019, doi: <http://dx.doi.org/10.30829/algoritma.v3i2.6440>.
- [8] M. Rossaria, B. Susilo, dan Ernawati, “Implementasi Algoritma Pencocokan String Knuth-Morris-Pratt dalam Aplikasi Pencarian Dokumen Digital Berbasis Android” *Rekursif: Jurnal Informatika*, vol. 3, no. 2, Nov. 2015, doi: <https://doi.org/10.33369/rekursif.v3i2.752>.
- [9] Anofrizen, “Perancangan Sistem Pencarian Nama Latin Tumbuhan Berbasis Website Menggunakan Algoritma Knuth Morris Pratt (KMP)” *Jurnal Sains, Teknologi dan Industri*, vol. 17, no. 2, pp. 59-64, Jun. 2020, doi: <http://dx.doi.org/10.24014/sitekin.v17i2.9769>.
- [10] H. T. Sadiyah dan M. S. N. Ishlah, “Implementation the Knuth Morris Pratt (KMP) Algorithm in Interactive Web Monitoring and Recording Rabbit Reproduction System” *Indonesian Journal of Artificial Intelligence and Data Mining*, vol. 2, no. 2, pp. 83-92, Sep. 2019, doi: <http://dx.doi.org/10.24014/ijaidm.v2i2.7411>.
- [11] D. Hermawaty, R. A. Krisdiawan, dan Y. Nurhayati, “Game Edukasi Word Search Puzzle Nama Anak Hewan Dalam Bahasa Sunda Menggunakan Algoritma Linear Congruent Method (LCM) Dan Algoritma Knuth Morris Pratt (KMP) (Studi Kasus: SDN Pajawanlor)” *JURNAL NUANSA INFORMATIKA*, vol. 13, no. 2, Jul. 2019, doi: <https://doi.org/10.25134/nuansa>.
- [12] F. T. Waruwu dan R. Mandala, “Perbandingan Algoritma Knuth Morris Pratt Dan Boyer Moore Dalam Pencocokan String Pada Aplikasi Kamus Bahasa Nias” *Jurnal Ilmiah INFOTEK*, vol. 1, no. 1, Feb. 2016.
- [13] M. Nurdin dan R. Ryoza, “Implementasi Algoritma Pencocokan String Knuth Morris Pratt Dalam Aplikasi Ensiklopedia Hadits Imam Bukhari Berbasis Web” *Jurnal Infomedia*, vol. 6, no. 2, Des. 2021, doi: <http://dx.doi.org/10.30811/jim.v6i2.2600>.
- [14] Habib dan I. H. Ikasari, “Penerapan Algoritma Knuth Morris Pratt Pada Fungsi Pencarian Arsip Dokumen Berbasis Web (Studi Kasus: Divisi Purchasing, PT Tifico Fiber Indonesia. TBK)” *Biner*, vol. 1, no. 2, pp. 76–86, Jun. 2023. [Daring]. Tersedia pada: <https://ejurnal.pnl.ac.id/infomedia/article/view/2600>.
- [15] H. T. Sa’diah, “Implementasi Algoritma Knuth-Morris-Pratt Pada Fungsi Pencarian Judul Tugas Akhir Repository” *Jurnal Komputasi*, vol. 14, no. 1, pp. 115-124, Jan. 2017, doi: [10.33751/komputasi.v14i1.262](https://doi.org/10.33751/komputasi.v14i1.262).