Penerapan Algoritma Boyer Moore Dalam Pencarian Barang Hilang pada Aplikasi FindIt Berbasis Android

Muhammad Afif Setiawan*, Septi Andryana, Aris Gunaryati

Fakultas Teknologi Komunikasi dan Informatika, Prodi Informatika, Universitas Nasional, Jakarta Selatan, Indonesia

Email: ^{1,*}20afif.setiawan@gmail.com, ²septi.andryana@civitas.unas.ac.id, ³arisgunaryati@civitas.unas.ac.id

Email Penulis Korespondensi: 20afif.setiawan@gmail.com

Abstrak— Kehilangan adalah salah satu peristiwa yang dialami oleh manusia dalam rentang hidupnya, dan ini adalah suatu hal yang wajar. Kehilangan barang tentunya dapat membuat seseorang gelisah dan panik, apalagi barang yang hilang berharga bagi dirinya. Maka dari itu dengan adanya perkembangan teknologi penulis mencoba membuat rancangan aplikasi yang dapat menjadi media informasi barang hilang bagi masyarakat, aplikasi ini dirancang berbasis Android dan diberi nama FindIt. Aplikasi FindIt ini menggunakan *Firestore Authentication* dan *Firestore Storage* sebagai *database* utama. Aplikasi ini dapat dijalankan secara *real-time* karena menggunakan *Realtime Firestore*. Pengujian algoritma dilakukan antara *Knuth Morris Pratt* dan *Boyer Moore*, dan hasilnya adalah *Boyer Moore* lebih cepat dalam melakukan pencocokan *string*. Hasil pengujian menggunakan algoritma ini adalah sebesar 100% akurat. Pengujian aplikasi dilakukan menggunakan metode *Blackbox*, dimana fungsi dan fitur pada aplikasi berjalan sesuai yang diharapkan. Dengan adanya aplikasi ini diharapkan masyarakat dapat saling membantu untuk mencari barang dan memposting barang temuannya tersebut pada aplikasi ini.

Kata Kunci: Aplikasi pencarian, *Boyer Moore*, *Knuth Morris Pratt*, Android, *Firestore*

Abstract—Losing is an experience that every human has experienced during his life span, and this is a natural thing. Loss of goods is an event that can make a person panic, especially if the lost item is valuable to him. Therefore, with the development of technology, the author tries to design an application that can be a medium for information on lost items for the community, this application is designed based on Android and is named FindIt. This FindIt app uses *Firestore Authentication* and *Firestore Storage* as the main database. This application can be run in real-time because it uses *Realtime Firestore*. Algorithm testing was carried out between *Knuth Morris Pratt* and *Boyer Moore*, and the result is *Boyer Moore* is faster in string matching. The test results using this algorithm are 100% accurate. Application testing is done using the *Blackbox* method, where the functions and features of the application run as expected. With this application, it is hoped that the community can help each other to find goods and post their findings on this application.

Keywords: Searching Application; *Boyer Moore*; *Knuth Morris Pratt*; Android; *Firestore*

1. PENDAHULUAN

Dewasa ini seiring dengan perkembangan jaman dan semakin banyaknya penduduk di Indonesia memungkinkan terjadinya kehilangan barang baik secara tidak sengaja maupun adanya tindak kriminal berupa pencurian. Di masyarakat, metode pencarian kehilangan barang sudah mulai berkembang, dimana tiap individu yang menemukan barang mempunyai keinginan untuk memberitahu informasi kehilangannya kepada orang lain, khususnya orang yang kehilangan. Terdapat sarana media informasi yang bisa digunakan untuk menginformasikan kehilangan, seperti televisi, koran, website, radio, dan media informasi lainnya. Dengan banyaknya informasi seperti itu, maka diperlukan sebuah wadah untuk menampung data dan informasi kehilangan secara yang terpusat agar dapat dimanfaatkan masyarakat untuk mengumumkan informasi dan mencari informasi kehilangan. Dan wadah tersebut adalah teknologi internet[1].

Dari permasalahan tersebut, seseorang yang ingin menyampaikan maupun mencari informasi tentang kehilangan membutuhkan suatu wadah berupa media yang lebih spesifik pada informasi kehilangan dan penemuan barang dan informasi dapat dengan mudah ditemukan melalui pencarian kata kunci. Dengan latar belakang tersebut maka penulis ingin membangun sebuah aplikasi FindIt, yang memiliki tujuan sebagai sarana informasi tentang barang hilang.

Banyak penelitian terdahulu yang sudah berusaha membuat aplikasi yang serupa salah satunya pada penelitian Agus Wantoro pada tahun 2018, dapat disimpulkan bahwa penemu barang akan lebih mudah untuk memberikan informasi kepada pemilik barang yang kehilangan dengan bantuan aplikasi semacam ini. Aplikasi ini dapat digunakan oleh masyarakat dan pihak yang berwajib untuk memberikan informasi kehilangan dan penemuan barang namun aplikasi harus dilanjutkan ke tahap implementasi sistem agar dapat digunakan, namun aplikasi perlu penambahan fitur-fitur supaya lebih memudahkan pengguna dalam menjalankan aplikasi ini[1].

Selanjutnya adalah penelitian Makrina dan Rivalri tahun 2017, meneliti aplikasi yang digunakan untuk mencari file teks pada komputer menggunakan algoritma *Knuth Morris Pratt* dan *Boyer Moore* untuk proses mencocokkan string supaya menemukan file teks tersebut pada komputer[2].

Lalu pada penelitian Agung Indrajaya tahun 2017 dari penelitiannya menjelaskan bahwa forum pelaporan dan penemuan barang hilang dapat membantu masyarakat pengguna internet untuk menyampaikan ataupun mencari barang hilang, dan juga barang yang ditemukan secara cepat dan terstruktur. Di lain sisi dapat dilakukan pencarian sesuai kategori maupun kata kunci agar mempermudah user dalam pencarian[3]. Lalu ada juga penelitian yang dilakukan oleh R Yu Tsarev pada tahun 2016. Penelitian tersebut mengimplementasikan pencarian string menggunakan algoritma *Knuth Morris Pratt* dan juga *Boyer Moore*, Algoritma ini KMP didasarkan pada

pencocokan pola maju dan Boyer-Moore didasarkan pada pencocokan pola mundur. hasil dari algoritma gabungan menunjukkan keuntungan dalam memecahkan masalah pencarian *string*[4].

Berdasarkan penelitian-penelitian tersebut, maka penulis membuat sebuah rancangan aplikasi berbasis Android yang penulis beri nama "FindIt" menggunakan algoritma *Knuth Morris Pratt* sebagai algoritma pencarian kata (*string*). Penulis bermaksud menambah fitur seperti mengirim pesan, login dan mendaftar serta fitur pencarian postingan. Diharapkan dengan adanya aplikasi ini maka dapat mempermudah pihak yang kehilangan maupun penemu barang untuk saling bertukar informasi di aplikasi FindIt. Pihak yang menemukan barang dapat memposting barang temuannya dan pihak yang kehilangan dapat mencari barang yang hilang dengan mencarinya pada fitur pencarian, jika ada maka pihak pencari dapat mengirim pesan ke pihak penemu barang.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Pada tahapan penelitian ini penulis menggunakan proses *waterfall*. *Waterfall* digunakan karena metode ini memiliki tahapan dan juga urutan proses secara berurutan dan berkelanjutan[5], Sistem *waterfall* melakukan pengembangan perangkat lunak secara sistematis dan berurutan mulai dari kebutuhan sistem, desain sistem, *coding*, pengujian perangkat, serta pemeliharaan perangkat. Kelebihan menggunakan proses *waterfall* adalah lebih terstruktur.

Gambar 1. Tahapan Penelitian

Berikut penjelasan dari gambar 1.

a. Pengumpulan Data

Pada tahap pengumpulan data dilakukan untuk memperoleh informasi bacaan yang dibutuhkan agar mencapai tujuan penelitian. Pengumpulan data yang dilakukan adalah dari jurnal, buku, dan website terkait penelitian yang akan dilakukan

b. Analisis Kebutuhan Sistem

Tahap analisis kebutuhan sistem adalah menentukan kebutuhan bagi seluruh sistem, Tahap *Requirements System* ini adalah menganalisis kebutuhan perancangan dari sisi *Hardware* dan juga *Software*[6] serta tulisan dari sumber tertentu yang dapat membantu dalam menentukan solusi permasalahan yang ada. Kebutuhan sistem pada aplikasi ini adalah :

1) Perangkat Keras (*Hardware*)

Laptop dengan spesifikasi:

- Processor Intel Core i7 7700-HQ
- GeForce GTX1050 RAM 8 GB
- HDD 1TB

Smartphone Android dengan spesifikasi:

- Android 9.0
- MediaTek Helio P60 1.99 GHz
- RAM 4GB

2) Perangkat Lunak (*Software*)

- Windows 10 Home 64-bit
- Android Studio 4.1

c. Pembuatan Desain Aplikasi

Pada Desain aplikasi dari penelitian yang dilakukan, pembuatan desain dari aplikasi FindIt ini menggunakan pemrograman XML pada Android Studio. Desain yang dibuat mulai dari logo aplikasi, tampilan menu *login*, *register*, *home*, *chat*, *profile* dan juga menu untuk menambah postingan.

d. Pengkodean

Aplikasi FindIt ini menggunakan Bahasa pemrograman Java pada Android Studio yang selanjutnya diimplementasikan menggunakan algoritma *Knuth Morris Pratt* sebagai cara untuk melakukan pencocokan

string[2]. Pada algoritma *knuth morris pratt* dalam mencari kata dilakukan dengan pencocokan string. Bila belum menemukan kecocokan maka tetap melakukan proses pencocokan dengan melakukan pergeseran.

e. Implementasi

Pada tahap ini merupakan implementasi dari algoritma *Boyer Moore* dan *Knuth Morris Pratt*. Kedua algoritma ini digunakan untuk pencarian *string*, setelah kedua aplikasi di implementasikan nantinya akan dilakukan perbandingan algoritma manakah yang melakukan pencarian lebih cepat. Algoritma ini nantinya akan digunakan pada fitur pencarian postingan di aplikasi FindIt.

f. Pengujian

Proses pengujian pada aplikasi FindIt ini menggunakan metode Blackbox. Pada Blackbox testing ini melakukan pengujian pada aplikasi seperti user interface (UI) aplikasi, fungsi dan fitur yang ada pada aplikasi, serta menguji kesesuaian alur[7]. Pengujian ini bertujuan agar mengetahui apakah aplikasi sudah berjalan dengan baik sesuai yang diinginkan dan apakah masih ada kesalahan atau kekurangan lainnya. Selain melakukan pengujian, penulis melakukan pengembangan sistem aplikasi yang telah dirancang agar dapat menciptakan hasil yang maksimal supaya aplikasi dapat berjalan dengan baik dan benar. Dan jika diperlukan akan ada perbaikan-perbaikan kecil pada aplikasi[8].

2.2 Algoritma Boyer Moore dan Knuth Morris Pratt

a. Boyer Moore

Algoritma *Boyer Moore* merupakan salah satu algoritma pencarian string. Algoritma ini dianggap sebagai algoritma yang paling efisien untuk pencarian string. Tidak seperti algoritma pencarian string yang lain, algoritma *Boyer Moore* mencocokkan karakter mulai dari sebelah kanan pattern. Ide dari algoritma ini adalah bahwa dengan memulai pencocokan karakter dari kanan, dan bukan dari kiri, maka akan lebih banyak informasi yang didapat. Berikut langkah-langkah pencarian string pada algoritma *Boyer Moore*:

- Jika teks pada *pattern* dengan teks pada *string* tidak cocok, maka akan menimbulkan (*mismatch*).
- Jika *pattern* dengan teks pada *string* ada yang cocok, maka algoritma akan memberitahukan penemuan tersebut di posisi ini[9].
- Pergeseran *pattern* di dasarkan dari tabel *Match Heuristic* (MH) dan *Occurrence Heuristic* (OH). Jika *string* yang sedang dibandingkan tidak terdapat pada *pattern* maka pergeseran dilakukan sebanyak jumlah karakter yang terdapat pada *pattern*, tetapi jika karakter yang tidak cocok tersebut terdapat pada *pattern*, maka pergeseran dilakukan berdasarkan tabel[2].

b. Knuth Morris Pratt

Algoritma KMP (*Knuth Morris Pratt*) adalah algoritma yang bekerja dengan cara melakukan pencocokan string[10]. Algoritma KMP cukup efisien dan cocok untuk semua jenis pencarian string[11]. Algoritma Knuth Morris Pratt bergerak dari kiri ke kanan untuk mencocokkan teks pattern dengan kata yang akan dicari (*string*), kondisinya adalah sampai dengan berikut[12], yaitu :

- Teks pada *pattern* dengan teks pada *string* tidak cocok, maka akan menimbulkan (*mismatch*).
- Jika *pattern* dengan teks pada *string* cocok, maka algoritma akan menyimpan informasi tersebut lalu melanjutkan ke *pattern* berikutnya.

3. HASIL DAN PEMBAHASAN

3.1 Penerapan Algoritma Boyer Moore

Algoritma *Boyer Moore* merupakan salah satu algoritma pencarian string. Algoritma ini dianggap sebagai algoritma yang paling efisien untuk pencarian *string*. Tidak seperti algoritma pencarian *string* yang lain, algoritma *Boyer Moore* mencocokkan karakter mulai dari sebelah kanan *pattern*. Penerapan algoritma *Boyer Moore* pada aplikasi adalah saat pengguna mencari judul postingan barang hilang. Fitur pencarian ini dapat ditemukan pada menu *Home* bagian atas aplikasi. Dengan adanya fitur pencarian ini diharapkan akan memudahkan para pengguna untuk mencari barang yang dicari.

Gambar 2. Fitur pencarian postingan

Pada gambar 2 pengguna mencari barang dengan kata “merah” maka akan muncul postingan yang terdapat pattern “merah” pada string “Tas Merah” yang merupakan judul postingannya. Langkah-langkah implementasi pada algoritma *Boyer Moore* adalah sebagai berikut :

Tabel 1. Nilai OH dan MH pada pattern

Pattern	M	e	r	a	h
Occurrence Heuristic (OH)	4	3	2	1	0
Match Heuristic (MH)	5	5	5	5	1

Tabel 1 merupakan penentuan nilai *Occurrence Heuristic* (OH) dan *Match Heuristic* pada pattern “Merah”. Tabel ini digunakan untuk menentukan berapa banyak pergeseran *pattern* jika karakter pada *string* dan *pattern* ada yang cocok.

Langkah 1:									
Posisi Teks	1	2	3	4	5	6	7	8	9
String	T	a	s		M	e	r	a	h
Pattern	M	e	r	a	h				
Langkah 2:									
Posisi Teks	1	2	3	4	5	6	7	8	9
String	T	a	s		M	e	r	a	h
Pattern				M	e	r	a	h	
Langkah 3:									
Posisi Teks	1	2	3	4	5	6	7	8	9
String	T	a	s		M	e	r	a	h
Pattern					M	e	r	a	h
Langkah 4:									
Posisi Teks	1	2	3	4	5	6	7	8	9
String	T	a	s		M	e	r	a	h
Pattern						M	e	r	a
Langkah 5:									
Posisi Teks	1	2	3	4	5	6	7	8	9
String	T	a	s		M	e	r	a	h
Pattern						M	e	r	a
Langkah 6:									
Posisi Teks	1	2	3	4	5	6	7	8	9
String	T	a	s		M	e	r	a	h
Pattern					M	e	r	a	h

Gambar 3. Implementasi pencarian string menggunakan algoritma Boyer Moore

Gambar 3 merupakan implementasi dari algoritma *Boyer Moore* untuk pencarian kata (*string*). Pada langkah pertama string “M” dan pattern “h” tidak cocok, namun karakter “M” terdapat pada pattern, maka pergeseran selanjutnya berdasarkan nilai dari tabel[13]. OH. Pada tabel OH karakter “M” memiliki nilai 4, maka *pattern* bergerak sebanyak 4 langkah.

Langkah kedua antara *string* dengan *pattern* mengalami kecocokan, maka tidak melakukan pergeseran dan informasi ini disimpan. Langkah selanjut melakukan pencocokan *pattern* kembali dan cocok. Proses pencocokan akan terus berlangsung sampai langkah keenam antara *string* dengan *pattern* cocok secara keseluruhan.

Tabel 2. Pengujian Akurasi Algoritma Knuth Morris Pratt pada String dan Pattern

No.	Input	Output	Hasil
1	Tas	Tas	Akurat 100%
2	merah	merah	Akurat 100%
3	Jaket	Jaket	Akurat 100%
4	perak	perak	Akurat 100%
5	Helm	Helm	Akurat 100%
6	Jam	Jam	Akurat 100%
7	hitam	hitam	Akurat 100%
8	dompet	dompet	Akurat 100%
9	kamera	kamera	Akurat 100%
10	Sony	Sony	Akurat 100%
11	kucing	kucing	Akurat 100%
12	hijau	hijau	Akurat 100%
13	warna	warna	Akurat 100%
14	tangan	tangan	Akurat 100%
15	Processor	Processor	Akurat 100%

Pada tabel 2 merupakan hasil pengujian dari pencarian kata di aplikasi FindIt, setiap kata yang ditulis akan muncul postingan yang mengandung kata tersebut. Di dapatkan tingkat akurasi kecocokan pencarian kata atau string menggunakan algoritma Boyer Moore pada postingan sebesar 100%.

3.2 Penerapan Algoritma Knuth Morris Pratt

Algoritma KMP (*Knuth Morris Pratt*) adalah algoritma yang bekerja dengan cara melakukan pencocokan *string*[10]. Algoritma KMP cukup efisien dan cocok untuk semua jenis pencarian *string*[11]. Penerapan algoritma KMP pada aplikasi ini adalah saat pengguna mencari judul postingan barang hilang. Fitur pencarian ini dapat ditemukan pada menu *Home* bagian atas aplikasi. Langkah-langkah implementasi pada algoritma KMP adalah sebagai berikut :

Gambar 4. Implementasi pencarian kata menggunakan algoritma *Knuth Morris Pratt*

Gambar 4 merupakan implementasi algoritma KMP (*Knuth Morris Pratt*) sebagai pencarian kata, Langkah pertama adalah mencocokkan *string* “T” dengan *pattern* “M” namun tidak cocok, maka *pattern* tersebut akan bergeser satu langkah ke kanan. Jika belum ada kecocokan antara *string* dengan *pattern* maka *pattern* tersebut akan terus bergeser ke kanan[14]. Pada Langkah kelima *string* “M” dan *pattern* “M” cocok, karena ada kecocokan teks maka algoritma *Knuth Morris Pratt* akan menyimpan informasi tersebut dan *pattern* tidak bergeser lalu melanjutkan proses pencocokan *string* dengan *pattern*. Sampai di Langkah sembilan karena jumlah *pattern* hanya 5 huruf dan *string* dan *pattern* sudah cocok maka proses selesai.

a. Hasil Perbandingan Algoritma *Boyer Moore* dan *Knuth Morris Pratt*

Berdasarkan penerapan dari dua algoritma yaitu *Boyer Moore* dan *Knuth Morris Pratt*, maka dapat ditentukan hasil perbandingannya. Pada algoritma *Boyer Moore* dilakukan pencocokan dari *string* “Tas Merah” dan *pattern* “Merah” sebanyak 6 langkah, sedangkan pencocokan menggunakan algoritma *Knuth Morris Pratt* adalah sebanyak 9 langkah. Maka dari itu untuk dari dua algoritma tersebut lebih baik menggunakan *Boyer Moore*[15] untuk mempersingkat waktu pencarian.

3.3 Diagram *Usecase* Aplikasi

Rancangan dari aplikasi FindIt ini adalah menggunakan *usecase*. *Usecase* diagram digunakan untuk menggambarkan skenario interaksi antara pengguna (*user*) dengan sistem. Fungsi *usecase* bertujuan untuk mendeskripsikan langkah-langkah proses yang dilakukan aktor terhadap sistem maupun sebaliknya[16]. Berikut merupakan *usecase* diagram aplikasi FindIt :

Gambar 5. Diagram *Usecase* Aplikasi

Pada gambar 5 merupakan *Usecase* diagram dari aplikasi FindIt, pengguna dapat melakukan register lalu login pada aplikasi. Setelah login, pengguna dapat membuat postingan barang hilang, mencari barang hilang dengan fitur pencarian, mengirim pesan ke pemosting, dan mengedit profil.

3.4 Tampilan Aplikasi

Dari penelitian yang dilakukan, pembuatan desain dari aplikasi FindIt ini menggunakan pemrograman XML pada Android Studio. Desain yang dibuat adalah menu login, register, menu home (menu utama), daftar chat, dan juga profil pengguna. Desain aplikasi dapat dilihat pada gambar 6 dan 7.

Gambar 6. Tampilan Aplikasi *Login, Register, Home, Chat*

Pada gambar 6 merupakan tampilan aplikasi FindIt, pada menu login pengguna harus sudah terdaftar pada aplikasi, jika belum mendaftar bisa klik register pada menu login. Setelah melakukan register dan login, maka pengguna akan masuk ke menu Home yang merupakan menu utama aplikasi ini. Pada menu Home merupakan tampilan postingan barang dari para pengguna yang sudah memposting di aplikasi FindIt. Lalu menu Chat merupakan daftar percakapan yang telah dilakukan, pengguna dapat melihat sudah melakukan percakapan dengan siapa saja di menu ini.

Gambar 7. Tampilan Aplikasi *Profile, Edit Profile, Tambah Postingan, Cari Postingan*

Pada gambar 7 merupakan tampilan menu lainnya pada aplikasi FindIt. Pada menu *Profile* terdapat fitur *edit profile* agar pengguna dapat mengubah foto *profile*, nama, dan nomor ponsel. Lalu ada menu tambah postingan untuk memposting barang temuan, pengguna memasukan foto, judul postingan, lokasi ditemukannya, nomor ponsel dan juga detail informasi. Pada menu pencarian postingan pengguna dapat mengetik kata yang ingin dicari lalu akan muncul postingan yang mengandung kata yang diketikkan, fitur ini tentunya sangat membantu para pencari barang di aplikasi ini.

3.5 Pengujian Aplikasi

Pada tabel 3 merupakan pengujian menggunakan metode *blackbox*. *Blackbox* testing merupakan proses pengujian suatu perangkat lunak yang mencakup bagian luar suatu aplikasi, mulai dari tampilan luar hingga proses *input* [17]. Pengujian mencakup menu *Home*, *chat*, *profile*, daftar *chat*, dan menu tambah postingan, hasil pengujian yang di dapat adalah “Berhasil”.

Tabel 3. Pengujian aplikasi menggunakan metode Blackbox

Kelas Uji	Cara pengujian	Hasil yang diharapkan	Hasil
Menu <i>Home</i>	Menampilkan daftar postingan	Dapat menampilkan semua postingan, mencari postingan, dan menekan tombol kirim pesan	Berhasil
Menu <i>Chat</i>	Tombol kirim pesan	Dapat mengirim pesan ke pengguna lainnya	Berhasil
Menu Daftar <i>Chat</i>	Tombol <i>chat</i>	Dapat menampilkan daftar <i>chat</i> pengguna	Berhasil
Menu <i>Profile</i>	Tombol <i>profile</i>	Dapat menampilkan informasi pengguna, dan dapat mengubah gambar serta data pengguna	Berhasil
Menu Tambah Postingan	Tombol tambah postingan	Dapat membuat postingan dan akan muncul di menu <i>Home</i>	Berhasil

4. KESIMPULAN

Aplikasi FindIt merupakan aplikasi media informasi pencarian barang hilang, aplikasi ini dirancang agar dapat membantu masyarakat dalam menemukan barang mereka, serta orang yang menemukan juga dapat memposting pada aplikasi tersebut. Serta terdapat fitur kirim pesan untuk memudahkan komunikasi antar pengguna. Aplikasi ini menggunakan *Firestore Storage*, *Firestore Authentication* dan *Realtime Database* sebagai database utama. Aplikasi ini dapat dijalankan secara *real-time* karena menggunakan *Realtime Firebase*. Pengguna dapat melihat postingan dan dapat mencari postingan menggunakan fitur cari postingan di menu *Home*. Pengguna juga dapat mengirim pesan ke pemosting agar memudahkan dalam komunikasi antar pengguna. Hasil dari pengujian algoritma *Boyer Moore* dan *Knuth Morris Pratt* adalah algoritma *Boyer Moore* lebih cepat dalam melakukan pencarian kata (*string*) pada fitur cari postingan, hasil pengujian pencarian adalah akurat 100%. Hasil pengujian menggunakan metode *Blackbox* adalah berhasil 100% dengan menguji menu *home*, *chat*, daftar *chat*, *profile*, dan juga tambah postingan. Maka dari itu adanya aplikasi ini diharapkan akan memudahkan pengguna dalam mencari barang yang hilang.

REFERENCES

- [1] A. Wantoro, “Prototype Aplikasi Berbasis Web Sebagai Media Informasi Kehilangan Barang,” *J. Teknoinfo*, vol. 12, no. 1, p. 11, 2018, doi: 10.33365/jti.v12i1.39.
- [2] M. M. Y. Daeli and R. K. Hondro, “Perancangan Aplikasi Pencarian Kata Dengan Kombinasi Algoritma Knuth Morris Pratt Dan Algoritma Boyer Moore,” *Maj. Ilm. INTI*, vol. 12, no. 2, pp. 271–275, 2017.
- [3] A. Indrajaya, “FORUM ONLINE SEBAGAI SARANA PENGADUAN KEHILANGAN DAN PENEMUAN BARANG HILANG,” pp. 1–6, 2017.
- [4] N. Khasanah, “PENERAPAN ALGORITMA KNUTH MORRIS PRATT PADA APLIKASI PENCARIAN BERKAS SHIPMENT BERBASIS WEB (Studi Kasus di PT YEC Semarang),” *E-bisnis*, vol. 11, pp. 14–22, 2018.
- [5] D. S. Purnia, A. Rifai, and S. Rahmatullah, “Penerapan Metode Waterfall dalam Perancangan Sistem Informasi Aplikasi Bantuan Sosial Berbasis Android,” *Semin. Nas. Sains dan Teknol.* 2019, pp. 1–7, 2019.
- [6] A. W. Saputra, W. A. Kusuma, and W. Suharso, “Rancang Bangun Aplikasi Pemesanan Molly Molen Malang Berbasis Android Menggunakan Metode Waterfall,” *J. Repos.*, vol. 2, no. 7, p. 855, 2020, doi: 10.22219/repositor.v2i7.763.
- [7] M. Nurudin, W. Jayanti, R. D. Saputro, M. P. Saputra, and Y. Yulianti, “Pengujian Black Box pada Aplikasi Penjualan Berbasis Web Menggunakan Teknik Boundary Value Analysis,” *J. Inform. Univ. Pamulang*, vol. 4, no. 4, p. 143, 2019, doi: 10.32493/informatika.v4i4.3841.
- [8] Y. Latif, “SISTEM PENCARIAN DAN PENGUMUMAN BARANG HILANG,” *IOSR J. Econ. Financ.*, vol. 3, no. 1, p. 56, 2016, doi: <https://doi.org/10.3929/ethz-b-000238666>.
- [9] C. P. Lestari, N. A. Hasibuan, and G. L. Ginting, “Perancangan Aplikasi Kamus Istilah Medis Berbasis Android Dengan Algoritma Boyer-Moore,” *J. INFOTEK*, vol. II, no. 3, pp. 1–6, 2016.

- [10] R. Rahim, I. Zulkarnain, and H. Jaya, "A review: Search visualization with Knuth Morris Pratt algorithm," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 237, no. 1, 2017, doi: 10.1088/1757-899X/237/1/012026.
- [11] H. L. Hendric Spits Warnars, J. Aurellia, and K. Saputra, "Translation Learning Tool for Local Language to Bahasa Indonesia using Knuth-Morris-Pratt Algorithm," *TEM J.*, vol. 10, no. 1, pp. 55–62, 2021, doi: 10.18421/tem101-07.
- [12] Nursobah and P. Pahrudin, "Penerapan Algoritma Pencarian Knuth-Morris-Pratt (Kmp) Dalam Sistem Informasi Perpustakaan Smk Ti Pratama," *Sebatik*, vol. 23, no. 1, pp. 112–115, 2019, doi: 10.46984/sebatik.v23i1.451.
- [13] A. Fau, Mesran, and G. L. Ginting, "Analisa Perbandingan Boyer Moore Dan Knuth Morris Pratt Dalam Pencarian Judul Buku Menerapkan Metode Perbandingan Eksponensial (Studi Kasus : Perpustakaan STMIK Budi Darma)," *J. Times (Technology Informatics Comput. Syst.*, vol. 6, no. 1, pp. 12–22, 2017.
- [14] R. Y. Tsarev, A. S. Chernigovskiy, E. A. Tsareva, V. V. Brezitskaya, A. Y. Nikiforov, and N. A. Smirnov, "Combined string searching algorithm based on knuth-morris- pratt and boyer-moore algorithms," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 122, no. 1, 2016, doi: 10.1088/1757-899X/122/1/012034.
- [15] F. T. Waruwu and R. Mandala, "Perbandingan Algoritma Knuth Morris Pratt dan Boyer Moore Dalam Pencocokan String Pada Aplikasi Kamus Bahasa Nias," *J. Ilm. INFOTEK*, vol. 1, no. 1, pp. 36–43, 2016.
- [16] D. Rosadi and F. O. Andriawan, "Aplikasi sistem informasi pencarian tempat kos di kota bandung berbasis android," vol. 10, no. 1, pp. 50–58, 2016.
- [17] T. Hidayat and M. Muttaqin, "Pengujian Sistem Informasi Pendaftaran dan Pembayaran Wisuda Online menggunakan Black Box Testing dengan Metode Equivalence Partitioning dan Boundary Value Analysis," *J. Tek. Inform. UNIS JUTIS*, vol. 6, no. 1, pp. 2252–5351, 2018.