



Aplikasi Pencarian Informasi Pondok Pesantren Se-Indonesia Dengan Algoritma Zhu-Takaoka Berbasis Android

Muhammad Hudan Adha, Abdul Sani Sembiring, Hukendi Hutabarat

Program Studi Teknik Informatika, STMIK Budi Darma, Medan, Indonesia
Email : ¹mhudan96@gmail.com, ²gurkiy@gmail.com, ³hukendikhutabarat@gmail.com

Abstrak

Pondok pesantren adalah suatu lembaga pendidikan agama Islam yang berbasis asrama. Sampai saat ini, Pondok Pesantren sudah memiliki tingkatan yang sama dengan SMP dan SMA. Sehingga banyak dari orang tua yang ingin anaknya sekolah di pesantren. Namun, sebagian orang tua kurang mendapat informasi dari pesantren tersebut. Seperti alamat, kegiatannya, dan sebagainya. Salah satu cara untuk mendapatkan informasi tersebut adalah dengan browsing di internet. Tapi itu hanya sebegaimana saja. Karena sebagian besar pondok Pesantren tidak memiliki website resmi. Dan juga untuk para calon alumni pesantren yang ingin menyelesaikan kewajiban terakhirnya dengan mengabdikan di Pondok Pesantren lain juga akan kesulitan untuk mencari informasi seputar Pondok Pesantren, mengingat tidak diperbolehkannya membawa barang elektronik ke Pesantren. Untuk mengatasi permasalahan di atas penulis ingin merancang sebuah aplikasi berbasis android dengan mengimplementasikan algoritma zhu-takaoka. Algoritma zhu-takaoka merupakan salah satu algoritma string matching yang dapat melakukan pencarian kata dengan pergeseran yang banyak yaitu dengan array 2 (dua) dimensi.

Kata Kunci: Pencarian, Informasi, Pondok Pesantren, Zhu-Takaoka, Android

Abstract

Islamic boarding school is a dormitory-based Islamic religious education institution. Until now, Islamic boarding schools have had the same level as junior and senior high schools. So that many of the parents want their children to go to a pesantren. However, some parents lack information from the pesantren. Such as address, activities, and so on. One way to get this information is by browsing on the internet. But that is only part of it. Because most Islamic boarding schools don't have official websites. And also for prospective Islamic boarding school graduates who want to complete their final obligations by serving at other Islamic boarding schools, it will also be difficult to find information about Islamic boarding schools, given that they are not allowed to bring electronic items to Pesantren. To overcome the above problems the author wants to design an android-based application by implementing the zhu-takaoka algorithm. Zhu-Takaoka algorithm is one string matching algorithm that can do word search with a lot of shifts with an array of 2 (two) dimensions.

Keywords: Search, Information, Islamic Boarding Schools, Zhu-Takaoka, Android

1. PENDAHULUAN

Dalam kehidupan sehari-hari informasi merupakan hal sangat dibutuhkan. Semua orang diseluruh dunia sudah tidak ada batasan lagi untuk dapat mengakses informasi apa saja dengan mudah. Dengan *smartphone* yang hanya mengandalkan sentuhan jari, *user* dapat dengan mudah mengetahui apa yang sedang terjadi di negara lain yang sangat jauh jaraknya. Dan Juga dapat dengan mudah mengakses informasi pribadi seseorang dengan memanfaatkan sosial media. Selain menerima informasi *user* juga dapat menyebarkan informasi dengan mudah ke seluruh dunia. Mulai dari bangun tidur sampai tidur kembali kehidupan selalu berhubungan dengan informasi.

Pondok Pesantren merupakan suatu lembaga pendidikan berbasis asrama yang khusus mempelajari ilmu agama Islam. Bagi masyarakat Indonesia, Pondok Pesantren sudah memiliki tingkat yang sama dengan SMP dan SMA di sekolah-sekolah lain pada umumnya. Dalam pembagiannya, Pondok Pesantren dibagi menjadi dua. Yaitu, Pondok Pesantren Modern dan *Salafiyah*. Perbedaan antara kedua Pondok Pesantren tersebut terletak pada pelajaran yang mereka pelajari. Walaupun sama-sama mempelajari ilmu agama, namun di Pondok Pesantren Modern ditambah dengan pelajaran umum seperti, Matematika, Bahasa Indonesia, Ekonomi, dan pelajaran umum lainnya. Berbeda dengan Pondok Pesantren *Salafiyah* yang hanya mempelajari ilmu agama saja. Di Indonesia sendiri keberadaan Pondok Pesantren sudah ada sejak zaman penjajahan.

Walaupun keberadaan Pondok Pesantren di Indonesia sudah ada sejak zaman penjajahan sebelum kemerdekaan Indonesia, namun masih sedikit informasi seputar Pondok Pesantren di Indonesia ini. Kurangnya informasi membuat para sebagian masyarakat khususnya orang tua yang ingin menyekolahkan anaknya ke Pondok Pesantren menjadi kesulitan. Jika mencari di *internet* dengan *search engine* hanya ada beberapa informasi saja yang tersedia. Dan juga memerlukan waktu yang lama. Dan akhirnya informasi tersebut hanya didapat dari orang lain yang belum tentu kebenarannya.

Dan juga bagi pihak pesantren yang ingin mengundang pesantren lain dalam sebuah acara, dimana acara tersebut ditugaskan oleh para santri. Dikarnakan para santri tidak diperbolehkan membawa alat elektronik, maka akan sulit untuk mengundang pesantren lain karena kurangnya informasi.

Algoritma pencocokan *string* (*String Matching*) merupakan salah satu bagian terpenting dalam berbagai proses yang berkaitan dengan data yang bertipe teks. Algoritma pencocokan *string* (*String Matching*) adalah algoritma untuk melakukan pencarian semua kemunculan string pendek (*pattern*) yang muncul dalam teks. *Pettern* yaitu string dengan panjang m karakter ($m < n$). Teks (*text*) yaitu long string yang panjangnya n karakter.



Berbagai perangkat lunak yang banyak digunakan dengan berbagai macam sistem operasi yang berbeda pula, menjadikan algoritma *string matching* sebagai dasar implementasi. Salah satunya adalah algoritma *Zhu-Takaoka*. Algoritma *Zhu-Takaoka* merupakan algoritma yang menggunakan dua karakter untuk menghitung pergeseran karakter berdasarkan *bad-character*. Untuk mencari ketidakcocokan atau mencari seluruh *text* menggunakan *preprocessing hashing*. Hal ini efektif untuk pencarian *string* dua dimensi [1].

Untuk mengatasi masalah tersebut, penulis ingin merancang sebuah aplikasi dengan perangkat *smartphone* yang berbasis sistem operasi android dan dapat digunakan secara luring (Luar Jaringan) untuk diimplementasikan dengan algoritma *zhu-takaoka*.

2. METODE PENELITIAN

2.1 Pondok Pesantren

Pesantren adalah sebuah pendidikan tradisional yang para siswanya tinggal bersama dan belajar di bawah bimbingan guru yang lebih dikenal dengan sebutan kyai atau ustadz dan mempunyai asrama untuk tempat menginap santri. Santri tersebut berada dalam kompleks yang juga menyediakan masjid untuk beribadah, ruang untuk belajar, dan kegiatan keagamaan lainnya. Kompleks ini biasanya dikelilingi oleh tembok untuk dapat mengawasi keluar masuknya para santri sesuai dengan peraturan yang berlaku. Pondok Pesantren merupakan dua istilah yang menunjukkan satu pengertian. Pesantren menurut pengertian dasarnya adalah tempat belajar para santri, sedangkan pondok berarti rumah atau tempat tinggal sederhana terbuat dari bambu. Di samping itu, kata pondok mungkin berasal dari Bahasa Arab *Funduq* (فندق) yang berarti asrama atau hotel. Di Jawa termasuk Sunda dan Madura umumnya digunakan istilah pondok dan pesantren, sedang di Aceh dikenal dengan istilah dayah atau rangkang atau meunasah, sedangkan di Minangkabau disebut surau.

2.2 String Matching

String matching atau disebut pencocokan *string*, adalah algoritma pencarian kemunculan *string* pendek. Pencocokan *string* (*string matching*) merupakan bagian penting dari sebuah proses pencarian *string* (*string searching*) dalam sebuah dokumen. Hasil dari sebuah pencarian *string* dalam dokumen tergantung dari teknik dan cara pencocokan *string* yang digunakan [5].

2.3 Algoritma zhu-takaoka.

Algoritma *Zhu-Takaoka* diciptakan oleh R.F.Zhu dan T.Takaoka pada tahun 1989. Algoritma ini menggunakan dua *text* karakter untuk menghitung pergeseran karakter berdasarkan *bad-character*. Untuk mencari ketidakcocokan atau mencari seluruh *text* menggunakan *preprocessing hashing*. Hal ini efektif untuk pencarian *string* dua dimensi [1].

Algoritma *Zhu-Takaoka* menjelaskan bahwa pencarian pattern dilakukan mulai dari tiap baris mulai dari baris ke 0 dan berakhir pada baris ke $n_1 - m_1$. Zhu dan Takaoka merancang sebuah algoritma yang dapat bekerja menggunakan pergeseran dengan *bad-character* untuk dua karakter *text* secara berurut.

Selama fase pencarian pencocokan karakter dilakukan dari kanan ke kiri dan ketika proses berada di posisi $y[j..j + m - 1]$ dan ketidakcocokan terjadi diantara $x[m - k]$ dan $y[j + m - k]$ ketika $x[m - k + 1..m - 1] = y[j + m - k + 1..j + m - 1]$ pergeseran dilakukan dengan pergeseran *bad-character* untuk *text* karakter $y[j + m - 2]$ dan $y[j + m - 1]$. Tabel *good-suffix* dari algoritma Boyer-Moore juga digunakan untuk menghitung pergeseran. Pada awal proses algoritma *Zhu-Takaoka* melakukan *preprocessing* untuk menghitung *badcharacter shift* dan *good suffix* dari algoritma Boyer-Moore [9].

3. ANALISA DAN PEMBAHASAN

3.1 Analisa Masalah

Analisa merupakan penguraian dari suatu pembahasan, analisa dilakukan guna mengidentifikasi dan mengevaluasi segala komponen dari aplikasi yang akan dibangun. Analisa merupakan tahap yang sangat penting dalam membangun sebuah aplikasi. Analisa dilakukan untuk mengevaluasi dalam permasalahan-permasalahan agar mendapatkan kekurangan dan keuntungan dalam aplikasi yang akan dibangun.

Berdasarkan masalah yang telah dibahas pada pendahuluan, didapat bahwa permasalahannya adalah bagaimana cara merancang aplikasi pencarian informasi pondok Pesantren dengan menggunakan salah satu Algoritma *string matching* yaitu Algoritma *Zhu-Takaoka* sebagai solusi. Adapun banyak pondok pesantren di Indonesia dan juga informasi seputar pondok pesantren tersebut, maka akan sulit untuk ditampilkan seluruh informasi pondok pesantren pada layar *smartphone* berbasis android yang terbatas. Belum lagi beberapa pondok pesantren yang memiliki nama yang sama. Akan memerlukan waktu yang lama untuk mencari informasi yang diinginkan.



Oleh karena itu, penulis membangun/merancang aplikasi pencarian informasi pondok pesantren dengan menerapkan Algoritma Zhu-Takaoka pada aplikasi tersebut. Informasi yang akan ditampilkan pun akan menjadi lebih sedikit karena dilakukannya proses seleksi berdasarkan kata kunci.

Aplikasi yang akan dirancang adalah aplikasi Informasi Pondok Pesantren dengan algoritma Zhu-Takaoka yang akan digunakan untuk pencocokan kata. Dalam aplikasi tersebut, akan menampilkan Informasi seputar Pondok Pesantren. Sumber kata yang digunakan sebagai pengujian yaitu berupa nama salah satu pondok pesantren di Indonesia.

Software yang digunakan untuk membangun aplikasi adalah *Eclipse Juno*. *Eclipse Juno* sendiri adalah IDE (*Integrated Development Environment*) yang sering digunakan untuk mengembangkan suatu *software* atau aplikasi yang dapat digunakan pada berbagai macam platform seperti windows, Lunix, MacOS, dan lain-lain. Eclipse Juno juga bisa menggunakan berbagai macam bahasa program. Dan untuk sistem operasi android, digunakan bahasa pemrograman JAVA.

3.2 Penerapan Algoritma Zhu-Takaoka

Algoritma *Zhu-Takaoka* adalah pengembangan dari Algoritma *Boyer-Moore* mempunyai ciri-ciri yang sama dalam proses pencocokan *string*. Yaitu fase *preprocessing* dan fase pencocokan. Perbedaan antara kedua algoritma tersebut adalah terletak pada tahap penentuan *bad character rule*. Dalam *Boyer-Moore*, bad character hanya terdiri *array* satu dimensi, sedangkan dalam *Zhu-Takaoka* dikembangkan menjadi *array* dua dimensi. Karakteristik algoritma *Zhu-Takaoka*, diantaranya:

1. Pengembangan dari algoritma *Boyer-Moore*.
2. Menggunakan *array* dua dimensi untuk menghitung nilai pergeseran.
3. Melakukan pencocokan dari kanan ke kiri.

Dalam pengerjaan algoritma zhu-takaoka, Setelah kata kunci atau *pattern* dimasukkan, langkah pertama adalah melakukan *preprocessing* yaitu mencari nilai *ztBc* (*Zhu-Takaoka Bad Character*) dan *ztGs* (*Zhu-Takaoka Good Suffix*) dari *pattern* yang telah dimasukkan. Setelah nilai tabel *ztBc* dan *ztGs* terisi semua, langkah selanjutnya adalah melakukan pencocokan, yaitu dengan cara membandingkan setiap karakter *pattern* dengan karakter pada teks mulai karakter paling kanan. Apabila pada perbandingan pertama sudah tidak ada kecocokan, maka nilai perpindahan yang digunakan adalah nilai perpindahan dari table *ztBc* dan *ztG* digunakan jika terjadi kecocokan pada *pattern* dan sumber teks. Proses perbandingan dilakukan sampai sisa sumber teks yang belum dibandingkan memiliki jumlah karakter kurang dari jumlah karakter pada *pattern*. Setelah proses perbandingan, langkah selanjutnya adalah informasi pondok pesantren dengan menggunakan algoritma *Zhu Takaoka*, baik pola itu ditemukan ataupun tidak ditemukan, dan proses pencocokan selesai.

Teks : PESANTREN MODERN MISBAHUL ULUM

PATTERN : MISBAHUL

Langkah pertama, membuat dan menentukan tabel *ztBc* (*Zhu-Takaoka Bad Character*) dan *ztGs* (*Zhu-Takaoka Good Suffix*) kemudian melakukan pencocokan.

1. Membuat tabel *ZtBc*

Langkah awal adalah menentukan tabel *ztBc*. Untuk memudahkan menentukan tabel *ztBc* adalah dengan menenukan tabel *BmBc* untuk mengidentifikasi setiap perpindahan. Berikut adalah tabel *BmBc*.

Tabel 1. *bmbc*

Index	0	1	2	3	4	5	6	7
Karakter	M	I	S	B	A	H	U	L
Nilai OH	7	6	5	4	3	2	1	8

Setelah mendapatkan nilai OH dari tabel *bmbc*, selanjutnya adalah membuat tabel *ztBc* dengan memasukkan semua karakter secara dua dimensi dengan nilai OH. Karakter diurutkan berdasarkan abjad. Seperti yang terlihat pada tabel berikut.

Tabel 2. *ztBc*

	A	B	H	I	L	M	S	U	*
A	8	8	2	8	8	7	8	8	8
B	3	8	8	8	8	7	8	8	8
H	8	8	8	8	8	7	8	1	8
I	8	8	8	8	8	7	5	8	8
L	8	8	8	8	8	7	8	8	8
M	8	8	8	6	8	7	8	8	8
S	8	4	8	8	8	7	8	8	8
U	8	8	8	8	8	7	8	8	8
*	8	8	8	8	8	7	8	8	8

2. Membuat tabel *ztGs*



Setelah mendapatkan hasil akhir pada tabel *ztBc*, maka dilanjutkan dengan membuat tabel *suffix* dari *pattern*. Berikut adalah tabel *ztGs*.

Tabel 3. *ztGs*

<i>Index</i>	0	1	2	3	4	5	6	7
<i>Karakter</i>	M	I	S	B	A	H	U	L
<i>Nilai MH</i>	8	8	8	8	8	8	8	1

3. Proses Pencocokan

Langkah terakhir adalah melakukan pencocokan *pattern* dengan sumber teks berikut:

Teks : PESANTREN MODERN MISBAHUL ULUM

Pattern : MISBAHUL

Tabel 4. Percobaan ke-1

<i>Index</i>	0	1	2	3	4	5	6	7	8	9	10	11	12
Teks	P	E	S	A	N	T	R	E	N		M	O	D
Pattern	M	I	S	B	A	H	U	L					

Pada percobaan ke-1 dapat dilihat bahwa karakter akhir pada *pattern* yaitu karakter L sejajar dengan karakter E pada sumber teks. Artinya, pada percobaan ke-1 terjadi ketidakcocokan. Maka dilakukan pergeseran sebanyak 8. Sesuai pada tabel *ZtBc* dengan pola R E.

Tabel 5. Percobaan ke-2

<i>Index</i>	6	7	8	9	10	11	12	13	14	15	16	17	18
Teks	R	E	N		M	O	D	E	R	N		M	I
Pattern			M	I	S	B	A	H	U	L			

Pada percobaan ke-2 dapat dilihat bahwa karakter akhir pada *pattern* yaitu karakter L sejajar dengan karakter N pada sumber teks. Artinya, pada percobaan ke-2 terjadi ketidakcocokan. Maka dilakukan pergeseran sebanyak 8. Sesuai pada tabel *ZtBc* dengan pola R N.

Tabel 6. Percobaan ke-3

<i>Index</i>	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Teks		M	O	D	E	R	N		M	I	S	B	A	H	U
Pattern								M	I	S	B	A	H	U	L

Pada percobaan ke-3 dapat dilihat bahwa karakter akhir pada *pattern* yaitu karakter L sejajar dengan karakter U pada sumber teks. Artinya, pada percobaan ke-3 terjadi ketidakcocokan. Maka dilakukan pergeseran sebanyak 1. Sesuai pada tabel *ZtBc* dengan pola H U.

Tabel 7. Percobaan ke-4

<i>Index</i>	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
Teks	N		M	I	S	B	A	H	U	L		U	L	U	M
Pattern			M	I	S	B	A	H	U	L					

Pada percobaan ke-4 dapat dilihat bahwa karakter akhir pada *pattern* yaitu karakter L sejajar dengan karakter L pada sumber teks. Artinya, pada percobaan ke-4 terjadi kecocokkan. Maka pencocokan dilakukan terhadap karakter sebelum karakter L, yaitu karakter U,H,A,B,S,I dan M pada *pattern*. Terjadi kecocokkan pada semua karakter. Karena jumlah pergeseran lebih besar dibandingkan dengan sisa karakter pada teks, maka pencocokan selesai.

4. IMPLEMENTASI

4.1 Kebutuhan Sistem

Kebutuhan sistem adalah perangkat yang digubakan oleh penulis dalam membangun aplikasi. Kebutuhan sistem terbagi 2 bagian. Software (perangkat lunak) dan Hardware (perangkat keras).

Perangkat lunak atau *software* merupakan komponen non fisik yang digunakan oleh penulis dalam membangun aplikasi informasi pondok pesantren. Adapun perangkat lunak yang digunakan dalam penelitian ini adalah:

1. Sistem Operasi *Windows 10*
2. *Eclipse Juno*
3. *Android Development Tool (ADT)*

4. Software Development Kit (SDK)

Perangkat keras atau hardware merupakan komponen fisik yang digunakan oleh penulis dalam membangun dan menguji aplikasi informasi pondok pesantren. Perangkat keras yang digunakan sebelumnya sudah di instal oleh perangkat lunak yang sudah disebutkan diatas. Adapun perangkat keras yang digunakan dalam penelitian ini adalah:

Laptop digunakan oleh penulis untuk membangun aplikasi informasi pondok pesantren. Adapun spesifikasi laptop yang digunakan dalam penelitian adalah:

Procesor : Intel Celeron
Memory : DDR3 2 GB
Harddisk : 500 GB
VGA : Intel HD Graphics 3000
Layar : 14 Inchi LED HD
Sistem Operasi : *Windows 10 Pro 64 bit*

Smartphone digunakan oleh penulis untuk menguji aplikasi informasi pondok pesantren. Adapun spesifikasi *smartphone* yang digunakan dalam penelitian adalah:

Layar : 5.5 inches
Sistem Operasi : *OS Android OS, v5.0.2 (Lollipop)*
Prosesor : *Octa-core 1.4 GHz Cortex-A53*
Memory : *Internal 16GB dan RAM 2GB*

Tampilan awal pada aplikasi terdiri dari tampilan *splashscreen*. *Splashscreen* adalah loading awal sebelum memasuki tampilan utama pada aplikasi. *Splashscreen* akan menampilkan gambar atau logo aplikasi.



Gambar 1. Screenshot SpashScreen

Tampilan pencarian merupakan tampilan yang disediakan pada tab Cari Informasi untuk mencari informasi pondok pesantren berdasarkan nama Pondok Pesantren. Tampilan pencarian menampilkan *EditText* dan *Button*. *EditText* berfungsi sebagai *tools* untuk menginput nama pondok pesantren yang akan dicari dan *Button* berfungsi sebagai tombol untuk melakukan pencarian. Adapun *screenshot* gambar untuk tampilan Tampilan *Searching* dapat dilihat pada berikut.



Gambar 2. Gambar Tampilan Pencarian

Tampilan informasi pondok pesantren adalah tampilan output atau tampilan hasil pencarian yang menampilkan informasi pondok pesantren berdasarkan kata kunci yang di input sebelumnya oleh pengguna. Adapun *screenshot* gambar untuk tampilan tampilan informasi pondok pesantren dapat dilihat pada gambar berikut.



Gambar 3. Tampilan Informasi Pondok Pesantren

Tampilan Option Menu adalah menu pilihan yang terdiri dari tiga menu. Yaitu, Bantuan, Tentang, dan Keluar. Adapun *screenshot* gambar untuk tampilan option menu dapat dilihat pada gambar berikut.



Gambar 4. Tampilan Option Menu

Tampilan Tentang adalah tampilan tentang penulis. Adapun *screenshot* gambar untuk tampilan option menu dapat dilihat pada gambar berikut.



Gambar 5. Tampilan Tentang

Tampilan Bantuan adalah tampilan bantuan cara menggunakan aplikasi.. Adapun *screenshot* gambar untuk tampilan option menu dapat dilihat pada gambar berikut.



Gambar 6. Tampilan Bantuan

5. KESIMPULAN

Bedasarkan hasil akhir dari penelitian ini, maka penulis dapat menarik kesimpulan dari penelitian “Aplikasi Pencarian Informasi Pondok Pesantren Dengan Algoritma Zhu-Takaoka Berbasis Android. Adapun kesimpulan dari penulis adalah:

1. Aplikasi ini dirancang untuk memudahkan pengguna dalam mencari informasi seputar pondok pesantren.
2. Dengan menerapkan algoritma zhu-takaoka pada aplikasi ini, pencarian kata yang dilakukan dengan dua kali proses menjadikan pencarian kata lebih efisien.
3. Aplikasi yang diuji berjalan dengan baik dan dapat melakukan pencarian kata dengan mudah dan efisien.

REFERENCES

- [1] P. J. & K.S.M.Panicker, “String Searching Algorithm Implementation-Performance Study with,” International Journal of Computer Science & Communication, vol. 1 No.2, p. 1, 2010.
- [2] Sutabri, Tata, Analisis Sistem Informasi, Yogyakarta: Penerbit ANDI, 2012.
- [3] Nawawi, "Sejarah dan Perkembangan Pesantren," IBDA', vol. 4, p. 1, 2006.
- [4] M. Dr.Suarga, Algoritma dan Pemograman, Yogyakarta: Penerbit ANDI, 2012.
- [5] E. Buulolo, "Implementasi Algoritma String Matching Dalam Pencarian Surat," Pelita Informatika Budi Darma, vol. 3, 2013.
- [6] D. Ardani Rahayu, “Implementasi Dan Analisis Algoritma Zhu-Takaoka Dan Algoritma Knuth-Morris-Pratt Pada Aplikasi Kamus Istilah Kesehatan Berbasis Android,” USU, 2017.
- [7] R. Munir, Algoritma & Pemograman Dalam Bahasa PASCAL dan C, Bandung : Informatika Bandung, 2011 (Edisi Revisi).
- [8] D. Knuth, Fundamental Algorithms, Boston: Addison-Wesley, 1973.
- [9] A. Kusnadi and A. K. Wicaksono, "Perbandingan Algoritma Horspool dan Algoritma Zhu-Takaoka dalam Pencarian String Berbasis Desktop," ULTIMA Computing, vol. IX, p. 14, Juni 2017.
- [10] Z. P. Juhara, Panduan Lengkap pemograman Android, P. S. Wibowo, Ed., Yogyakarta: Penerbit ANDI, 2016.
- [11] H. Jogiyanto, Sistem Teknologi Informasi, Yogyakarta: ANDI, 2003.
- [12] A. Rossa and M. Shalahuddin, Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek, Bandung: Informatika Bandung, 2013.
- [13] P. P. a. H. Widodo, Menggunakan UML, 1 ed., Bandung: INFORMATIKA, 2011.
- [14] B. Hariyanto, Esensi-esendi Bahasa Pemograman Java (Revisi Keempat), Bandung: Informatika Bandung, 2014, p. 1.
- [15] S. Rachmad Hakim, Mastering Java, Jakarta: PT. Elex Media Komputido, 2009.
- [16] Fathansyah, Basis Data, Bandung: Informatika Bandung, 2015 (Revisi Kedua).
- [17] A. A. B. Ginting and D. P. Utomo, "PERANCANGAN APLIKASI CATALOG WISATA DI SUMATERA UTARA MENGGUNAKAN ALGORITMA RABIN-KARP," KOMIK (Konferensi Nasional Teknologi Informasi dan Komputer), vol. 3, no. 1, pp. 57-63, 2019.