The 2nd International Conference on Multidisciplinary Engineering (2nd ICOMDEN 2024)

The 2nd International Conference on Multidisciplinary Engineering: 00030 (2024) https://doi.org/10.29103/icomden.v2.xxxx
Volume 2, 2024
eISSN: xxxx-xxxx
Research Original Article

Implementation of Horspool Algorithm on Book Search Application in Malikussaleh University Library Based on Mobile Android

Gilang Wahyu Ramadhan[1] Zara Yunizar[2] Sujacka Retno[3]
[1]Department Of Informatics, Universitas Malikussaleh, Bukit Indah, Lhokseumawe, 24353, Indonesia, gilang.200170245@mhs.unimal.ac.id
[2]Department Of Informatics, Universitas Malikussaleh, Bukit Indah, Lhokseumawe, 24353, Indonesia, zarayunizar@unimal.ac.id
[3] Department Of Informatics, Universitas Malikussaleh, Bukit Indah, Lhokseumawe, 24353, Indonesia, sujacka@unimal.ac.id
✉Corresponding Author: **gilang.200170245@mhs.unimal.ac.id | Phone: +6289668560602**

Abstract
The development of information technology encourages innovation in library management systems, one of which is an efficient book search system**.** This thesis examines the application of the *Horspool* method in library book search applications to improve search speed and accuracy. *The Horspool* method is a pattern matching algorithm designed to speed up the text search process by utilizing a sliding table, which significantly reduces the number of comparisons required in pattern search. The developed application allows users to search for books based on title, author, or other keywords with fast and relevant results. An evaluation was conducted by comparing the search time between the *Horspool* method and the traditional search method. The evaluation results show that the *Horspool* method offers significant performance improvement, with faster search time and high accuracy.
Keywords: : Book Search Application, Library, *Horspool* Method, Library Management System.

Introduction
The development of information and communication technology has brought changes in various sectors, including the library sector. The utilization of information technology as a means of improving the quality of services and operations has brought great changes in the library world. One example of the application of information technology can be seen in the application of *digital* libraries. Digitization is the process of converting printed documents into digital documents. [1] Digitalization is a new phenomenon that is starting to get a lot of attention, because several aspects of the benefits can be obtained from the speed of accessing information to the accuracy of the results (accuracy) of each data and information search, especially for the library field.

The library is one of the places that functions as a center of information, a source of knowledge, research, recreation, and the provision of various services for all people. [2]. Libraries at a university, especially at Malikussaleh University, are needed to support the implementation of education and teaching, research and community service. Increasing the amount of data on library material collections every year is a necessity to improve the quality of library services. The addition of new library material titles must be recorded and archived into the library material collection data to facilitate the search process. In addition, the placement of library materials must be neatly organized and arranged according to indexing so that it is easy to find when lecturers or students want to read.

Catalogs are visual communication media that contain complete information about products and other information for consumers to know. [3]. Book catalogs can include thousands or even millions of book entries. In a scenario like this, an application with a fast and efficient search method is needed so that users can easily find the desired book without having to wait for a long search time, therefore the *Horspool* algorithm is needed to make it easier.

Horspool algorithm is one of the algorithms that can be used to help solve problems with character or text-based search processes (string matching) with a simple way of working. [4]. This *Horspool* algorithm can be used to implement a quick search in a book catalog. When the user inputs search keywords, this algorithm can quickly find books that match the entered pattern .

The use of online catalogs makes it easier for students or lecturers to find data on library material collections without

being limited by space and time. Smartphone technology that is currently developing allows users to access information on a mobile basis. The ease and breadth of accessing information is the reason smartphone users are increasing every day. Many applications have been developed to support the performance of smartphones in making it easier for users to find the information they need. The development of this technology allows the creation of catalog applications by utilizing smartphone media, to access library information in this case a mobile catalog.

Related Works

1. Library

A library is an institution that manages collections of written works, printed works, and/or recorded works professionally with a standardized system to meet the educational, research, preservation, information, and recreational needs of the users. Library as one part of the place of education has a very important role to add insight and knowledge of students. Various sources of scientific information, books, literature from all types of library media, can be disseminated with a certain system [5].

2. Mobile Applications

Mobile applications are software that runs on mobile devices such as smartphones or tablet PCs. Mobile applications are also known as applications that can be downloaded and have certain functions that add to the functionality of the mobile device itself. To get the desired mobile application, users can download it through certain sites according to the operating system they have. Google Play and iTunes are some examples of sites that provide a variety of applications for Android and iOS users to download the desired application [6].

3. Android

Android is a collection of software on mobile devices that includes an operating system, middleware, and major mobile applications. Android is a Linux-based operating system, currently Android is transformed into a platform that innovates quickly thanks to its main product developer, Google, which acquired Android Pafda in 2009. Until now, Android has always released the latest version of its operating system, starting from Android version 1.1 which was released in March 2009, to Android 12 (*Snow Cone*) which was launched in October 2021. Android version 12 was first released on October 4, 2021, this version has a major update to Material Design which is then referred to as "*Material You*" Some of the advantages of this version are that the operating system can automatically generate color themes for the menu system and supported applications using the user's wallpaper color, equipped with features for taking screenshots in full or scrolling web views, and can prevent applications from using applications and microphones through quick settings [7].

4. Horspool Algorithm

Horspool's algorithm is one of the *string-matching* algorithms for finding a *pattern* in text, introduced and published by R Nigel Horspool in 1980. The problem with this text search is that the text searched can be very large (possibly hundreds of thousands of characters) so it is important to use a more efficient technique. Horspool's algorithm works with almost the same method as the Boyer-More algorithm but does not perform character-based jumps on petterns that are found not to match in the text. [8]. Horspool algorithm has the rightmost character shift value of the window. In the preprocessing stage, the shift value will be calculated for all characters. At this stage, petterns are compared from right to left until a pettern match or mismatch occurs. The rightmost character in the window is used as an index in performing the shift value. In case a mismatch (character not in the pattern) occurs, the window is shifted by the length of the pattern. Otherwise, the window is shifted by the rightmost character in the pattern [9].

There are two stages in string matching using the *Horspool* algorithm, namely:

1. Preprocessing In this stage, *pattern* observation is performed on the text to build a *bad-match* table that contains shift values when there is a mismatch between patterns Systematically, the steps performed by the Horspool algorithm in the preprocessing stage are:
    a. The Horspool algorithm matches the rightmost character in the pattern.
    b. Each character in the pattern is added to the bad-match table and its shift value is calculated.
    c. The character at the end of the *pattern* is not counted and is not used as the closest character of the same character as it.
    d. If there are two characters that are the same and one of them is not the rightmost character, then the character with the largest index has its shift value calculated.

The Horspool algorithm stores the length of the pattern as the length of the shift value by default if the characters in the text are not found in the pattern.

2. Search stage Systematically, the steps that the Horspool algorithm takes in the preprocessing stage are :
    a. A comparison of the rightmost character of the *pattern* against the window is performed.
    b. The bad-match table is used to skip characters when a mismatch occurs.
    c. When there is a mismatch, the rightmost character in the window serves as the basis for determining the shift distance to be performed.
    d. After matching either the result is matched or not matched, the window is shifted to the right.

This procedure is repeated until the window is at the end of the text or when the pattern matches the text .

Materials & Methods

1. System Design

At this stage the researcher designs an application system to input book data using the horspool method. The initial stage is to use DFD to explain more clearly the process that will be carried out in the system to be created.

2. System Implementation

System implementation refers to the stage where the application system design that has been designed previously is realized into a functioning entity after planning and analysis. Not only that, design implementation involves converting the design into real form.

3. System Testing

The final stage is application testing, which evaluates the quality of software by identifying errors or bugs, and verifying whether the application can meet software needs.

4. System Schematic

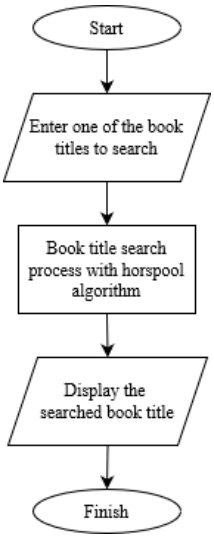

Figure 1. System Schematic

The system scheme above illustrates the book title search process using the Horspool algorithm. The process starts when the user enters one of the book titles that they want to search. After the title input is received, the system will perform a search based on the Horspool algorithm, which is an efficient string matching algorithm. Once the search process is complete, the searched book title will be displayed to the user. The last step of the process is to complete the operation after the searched book is successfully found and displayed.

Results and Discussion

1. Research Results

The application of the Horspool algorithm in this application is used for the process of searching book data in the library. The first step is to create a database using Firebase, with data containing books in the Malikussaleh University library. Then, the pattern search in the application will be processed through the database according to the Horspool algorithm search mechanism. If the pattern searched matches the input, then the search results from the database will be displayed in the form of book data and details. However, if the pattern is not found in the database, a statement will appear that the word is not found.

A. Use Case Diagram

Use case diagram is a modeling for the behavior of the system to be built, and describes an interaction between one or more actors with the system. Use case diagrams are also used to determine the functions that exist inside. The following is a use case diagram for the Library Book Search application system using the Horspool method.

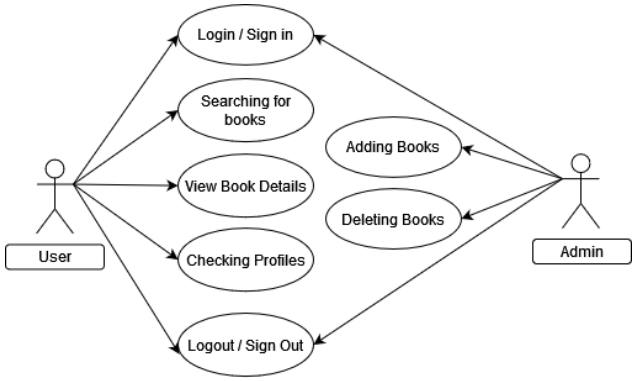

Figure 2. Use Case Diagram

The use case diagram above shows the interaction flow diagram between users and admins in the book management system. Users can login, search for books, view book details, check their profile, and logout. Meanwhile, the admin has the ability to add and delete books from the system. This diagram clearly illustrates the roles and functions of each user in managing and accessing book information.

B. Horspool Method Calculation

Pattern: WEB

W	E	B
0	1	2

Table 1. Bad-matches in preprocessing

Character	Index	Value
W	**0**	**5**
E	**1**	**4**
*	**2**	**3**

To find the shear value in the *Bad-match* Table, the following equation is used:

Value = $m - i - 1$

Description:

m : *pattern* length

i : index

Value = 3 - 0 -1 = 2

Value = 3 - 1 -1 = 1

* : unrecognized character

To illustrate the details of the algorithm, an example case will be given where pattern P = "WEB" and text T = "PEMOGRAMAN WEB SERI PHP" Initial initialization and bad-match generation.

Table 2. Bad-match initialization

m	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
P	P	E	M	O	G	R	A	M	A	N	_	W	E	B	_	S	E	R	I	_	P	H	P
T	W	E	B																				
i	0	1	2																				

Table 3. Bad-match Generation

P	W	E	*
I	0	1	2
v	2	1	3

As seen in Table 3 above, the initial bad-match initialization is performed. Each text and *pattern* is assigned a value *m* and *i* respectively, where *m is* the length of the pattern and *i* is the index. Table 3 shows the bad-match shift value by calculating the value of *v* as done in Table 3. In the initial stage of the search, the rightmost character of the *pattern is* compared against the window. If there is a mismatch, a right shift will be made to skip the mismatched character where the shift value is listed in the *bad-match* table. The rightmost character of the text in the *window* serves as the basis for determining the distance to be shifted. This can be seen in Table 4 below:

Table 4. First iteration of Horspool algorithm

m	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
P	P	E	M	O	G	R	A	M	A	N	_	W	E	B	_	S	E	R	I	_	P	H	P
T	W	E	B																				
i	0	1	2																				

In the first iteration of the table, there is a mismatch between the characters "M" and "B". The character "M" is not in the bad-match table so it is replaced by a sign (*). The sign (*) is worth 3 so it is done 3 times. This can be seen in Table 5.

Table 5. Second iteration of Horspool algorithm

m	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
P	P	E	M	O	G	R	A	M	A	N	_	W	E	B	_	S	E	R	I	_	P	H	P
T				W	E	B																	
i				0	1	2																	

In the second iteration of the table, there is another mismatch between the characters "R" and "B". The character "R" is not in the bad-match table so it is replaced by a sign (*). The sign (*) is worth 3 so it is done 3 times. This can be seen in Table 6.

Table 6. Third iteration of Horspool algorithm

m	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
P	P	E	M	O	G	R	A	M	A	N	_	W	E	B	_	S	E	R	I	_	P	H	P
T							W	E	B														
i							0	1	2														

In the third iteration of the Table, there is another mismatch between characters "A" and "B". The character "A" is not in the bad-match table so it is replaced by a sign (*). The sign (*) is worth 3 so it is done 3 times. This can be seen in Table 7.

Table 7. Fourth iteration of Horspool algorithm

m	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
P	P	E	M	O	G	R	A	M	A	N	_	W	E	B	_	S	E	R	I	_	P	H	P
T										W	E	B											
i										0	1	2											

In the fourth iteration seen in table 4.4, the pattern and text mismatch occurs in the characters "W" and "B". The character "W" is in the bad-match table which has a value of 2 so that it is shifted 2 times. This can be seen in Table 8.

Table 8. Fifth iteration of Horspool algorithm

m	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
P	P	E	M	O	G	R	A	M	A	N	_	W	E	B	_	S	E	R	I	_	P	H	P
T												W	E	B									
i												0	1	2									

In this fifth iteration, the window is at the end of the text and all patterns match the text. All character matching using the horspool algorithm has been completed and stopped at the fifth iteration.

2. System Testing

Table 9. System Working Test

No	Pattern	Number of words found	Running Time
1	Program	9	0,002
2	Web	11	0,002
3	Java	1	0,001
4	Photoshop	14	0,003
5	Desain	7	0,002
6	Video	3	0,001
7	Grafis	2	0,001
8	Administrasi	41	0,003
9	Organsasi	13	0,002
10	Pendidikan	19	0,003
11	Internasional	16	0,002
12	Adobe	3	0,001
13	Psikologi	6	0,001
14	Perbankan	21	0,002
15	Bank	40	0,003
16	Manajemen	153	0,003
17	Microsoft	2	0,001
18	Perhotelan	2	0,001
19	Hukum	70	0,003
20	Politik	58	0,002
21	Ekonomi	93	0,003
22	Indonesia	65	0,003
23	Komunikasi	53	0,002
24	Ilmu	50	0,002
25	Industri	5	0,001
26	Sosiologi	21	0,001
27	Teknik	14	0,002
28	Arsitektur	2	0,001
29	Spesial	1	0,001
30	Sahabat	1	0,001
31	Inter	26	0,002
32	Uang	45	0,003

33	Prinsip	5	0,001
34	Pemasaran	26	0,002
35	Statistik	14	0,001
36	Bisnis	28	0,002
37	Penggalian	1	0,001
38	Data	17	0,001
39	Python	3	0,001
40	Pandas	1	0,001
41	Desa	10	0,001
42	AutoCad	8	0,001
43	3ds	2	0,001
44	Regresi	1	0,001
45	Membaca	1	0,001
46	Riset	10	0,001
47	Metode	27	0,002
48	Pemustaka	1	0,001
49	Skripsi	3	0,001
50	Tesis	4	0,001
Total			**0,082**
Average			**0,001**

The table shows the number of words found in some tests and the *running time* taken in seconds. There are 50 *Patterns* that each record the number of words and the time taken to run the test. The average execution time of all tests was 0.001 seconds, with the total execution time reaching 0.082 seconds.

3. System Implementation

Figure 3. Splash Screen Display

The initial display of the application in the image shows the Malikussaleh University logo with a simple and clean design. The logo is dominated by green and gold colors, with the word "1979" in the center, indicating the year the university was founded. At the bottom of the logo, there is the text "Malikussaleh University" written in a formal and elegant font style. The logo on this application also indicates that all book data in the application is book data that the admin input at the library at Malikussaleh University.

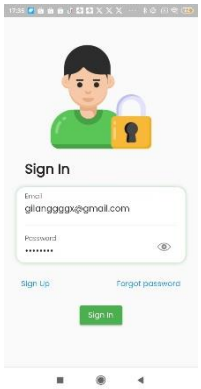

Figure 4. Login View

The following app view shows a simple and easy-to-understand *Sign In* page. At the top, there is an illustration of a user icon with a padlock, which signifies security in the sign-in process. Users are required to enter their email address and password in the two fields provided. In addition, there is a "*Sign Up*" option for new users and a "*Forgot password*" option to assist with account recovery if the user forgets their password. A green "Sign In" button at the bottom provides access to the app once the information entered is verified. This interface design prioritizes ease of use with clean and intuitive visual elements.

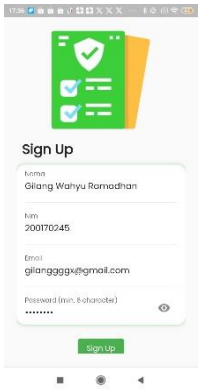

Figure 5. Sign up view

The *sign-up* view of this application displays a simple and intuitive registration page, there is a registration form with fields for "Name," "Nim," "Email," and "Password," each of which has been filled in with sample information. The name entered is Gilang Wahyu Ramadhan, the Nim is 200172045, the email provided is gilanggggx@gmail.com, and the password field shows characters hidden for security. At the bottom of the form, there is a green button labeled "Sign Up." This reflects a common step in accessing new software or services that require user registration.

Figure 6. Alert View

The app's interface features a pop-up notification with a yellow header and exclamation mark icon, as well as the text "ATTENTION!" that draws the user's attention. The message in this pop-up instructs the user to return the book to the shelf immediately after finishing reading. Below this message, there is an "Ok" button to close the notification.

Figure 7. Book Data Display

This application view displays a list of books in the library "Malikussaleh Library." Each book entry includes the book title, author, year of publication, and the associated library branch. For example, the book "Computer Networks" by Adam Sahil S.Si., was published in 2023 at Univ Reuleut Library. There is also the book "Photoshop Special Manipulation Techniques" by Insaf Idul Khoiri, published in 2023 at the same library, as well as "Using AutoCad" by A. Taufiq Hidayatullah, published in 2012. To the right of each book entry, there is a green button that serves to interact with the book list, such as getting more information about the book. At the top of the interface, there is a search bar that makes it easy for users to search for books within this library application.

Figure 8. Search View

The app's interface features an intuitive and easy-to-use search function. At the top, there is a search bar with the text "*Search by Title*" . Below that, the search results are displayed in green text on a white background. Each search result includes various course titles related to programming, Photoshop, AutoCAD, and 3D modeling. Interesting course titles such as "23 Special Techniques for Mastering Photoshop," "30 Seconds Guaranteed Entry to Heaven AutoCad," and "60 Minute Study for Beginners Photoshop CS4" indicate that this book title offers a variety of learning materials tailored to different skill levels and interests in technology. This look also reflects how the app can organize and display search results in an attractive and informative way, making it easy for users to find relevant courses quickly.

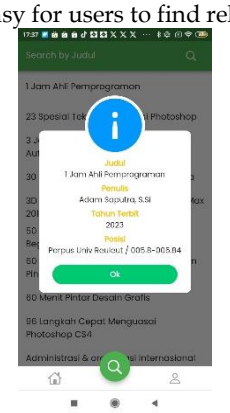

Figure 9. Book Detail View

The app's interface features an informative and easy-to-read book detail *pop-up*. This *pop-up* includes important information about the book such as title, author, year of publication, and the book's position in the library. For example, a book titled "Programming Algorithms" is written by Adam Saputra, S.Si., published in 2023, and its position is in the Releut University Library with the code 005.8-005.84. At the bottom of the *pop-up*, there is an "Ok" button to close the *pop-*

up. The design and layout of this detail window is clear and simple, making it easy for users to find the information they need quickly.

Figure 10. Profile View

In the app's profile view, users are greeted with a green banner at the top that reads "*Profile*" in white text. Below that, there is a circular *placeholder* for the user's profile photo. The user's name, "Gilang Wahyu Ramadhan," is clearly displayed below the profile picture. Next, there are two columns of information: one for "Nim" with the number "200170245," and another for "Email" with the address "gilanggggggx@gmail.com." At the bottom of the screen, there is a green "*Logout*" button that allows users to easily *log* out of their account.

Figure 11. Logout Display

In this view of the application, the user will see a logout confirmation screen. At the top, there is a pop-up with a large question mark symbol at the top, followed by the text "Are you sure you want to *Logout*?" asking if the user is sure they want to log out. Below this question, there are two buttons: one labeled "Cancel" with a red background and another labeled "Ok" with a green background, giving the option to cancel or confirm the logout process.

Conclusions

Based on the research results, the application of the *Horspool* method to the library book search application shows that this algorithm is very efficient and appropriate in searching for text patterns in the book database. Application testing shows that the Horspool algorithm is able to provide fast search results, especially when dealing with large databases. Moreover, in scenarios where the pattern or keyword entered by the user is not found in the database, the algorithm is able to quickly identify the condition and display an appropriate message, without causing any degradation in application performance.

The speed and efficiency of the *Horspool* method was also demonstrated in handling the case of an empty search, where no data matched the user input. The application remained stable and responsive, demonstrating that the algorithm is capable of handling such situations without introducing errors or glitches. Overall, the application of the *Horspool* algorithm provides significant benefits in improving search performance in the library system, making the search process faster and more reliable. This proves that the method is suitable for library applications that focus on managing and searching large amounts of data, thus providing an optimal and efficient user experience.

References

[1] Mustofa, "Digitalisasi Koleksi Karya Sastra Balai Pustaka sebagai Upaya Pelayanan di Era Digital Natives," *JPUA J. Perpust. Univ. Airlangga Media Inf. dan Komun. Kepustakawanan*, vol. 8, no. 2, p. 60, 2020, doi: 10.20473/jpua.v8i2.2018.60-67.

[2] R. Meiyanti, "Rancangan Aplikasi Perpustakaan Berbasis Android Di Perpustakaan Universitas Malikussaleh," *J.*

Teknol. Terap. Sains 4.0, vol. 3, no. 2, p. 783, 2022, doi: 10.29103/tts.v3i2.8275.

[3] P. U. D. Satrio, "Perancangan Katalog Wisata Kota Surabaya Sebagai Media Informasi Massa," *Gestalt*, vol. 1, no. 1, pp. 95–102, 2019, doi: 10.33005/gestalt.v1i1.23.

[4] I. Rofi'ah, K. Hantoro, and Mugiarso, "Perancangan Sistem Informasi Penjualan Obat-Obatan Berbasis Web Pada Apotek Diana Menggunakan Algoritma Horspool," *J. Students' Res. Comput. Sci.*, vol. 3, no. 2, pp. 195–206, 2022, doi: 10.31599/jsrcs.v3i2.1404.

[5] F. Nugraha, "Analisa Dan Perancangan Sistem Informasi Perpustakaan," *Simetris J. Tek. Mesin, Elektro dan Ilmu Komput.*, vol. 5, no. 1, pp. 27–32, 2014, doi: 10.24176/simet.v5i1.132.

[6] M. Irsan, "Rancang Bangun Aplikasi Mobile Notifikasi Berbasis Android Untuk Mendukung Kinerja Di Instansi Pemerintahan," *J. Penelit. Tek. Inform.*, vol. 1, no. 1, pp. 115–120, 2015, [Online]. Available: http://jurnal.untan.ac.id/index.php/justin/article/view/9984/9752

[7] Z. Yunizar, N. Savira Ersa, Z. Ardian, R. Rusnani, and F. Maulana Helmi, "'WASTEAPP' Application Based on Android for Household Waste Self-Tracking," *Int. J. Eng. Sci. Inf. Technol.*, vol. 2, no. 4, pp. 61–69, 2022, doi: 10.52088/ijesty.v2i4.327.

[8] V. Frigustini, A. Erlansari, and D. Andreswari, "IMPLEMENTASI ALGORITMA HORSPOOL PADA APLIKASI KAMUS BAHASA LINTANG-INDONESIA BERBASIS ANDROID," 2018. [Online]. Available: http://ejournal.unib.ac.id/index.php/rekursif/88

[9] W. Fuadi, R. Risawandi, and R. Yanti, "Aplikasi Doa Para Nabi Dan Rasul Dalam Al-Qur'an Menggunakan Algoritma Horspool Berbasis Android," *TECHSI - J. Tek. Inform.*, vol. 12, no. 1, p. 1, 2020, doi: 10.29103/techsi.v12i1.1702.

[10] D. Zaliluddin, B. Budiman, and A. Rully, "Implementasi E-Government Berbasis Android," *JSiI (Jurnal Sist. Informasi)*, vol. 7, no. 2, pp. 83–88, 2020, doi: 10.30656/jsii.v7i2.2052.