INTEGRATED Journal of Information Technology and Vocational Education 4(2) (2022) 57-118

INTEGRATED Journal of Information Technology and Vocational Education

Journal homepage: https://ejournal.upi.edu/index.php/integrated

Application of the boyer moore method in the application dictionary of web-based information technology terms (case study: pt. erefka tiga pilar utama)

Arini[1,], Rikza Muntazari[2], Hendra Bayu Suseno[3]*

[1,2,3]Information Technology, University Syarif Hidayatullah Jakarta, Indonesia
*Correspondence: E-mail: arini@uinjkt.ac.id

ABSTRACT

The emergence of many new terms in information technology because of the rapid development of technology makes people, especially those who pursue the IT field, must know and learn terms related to information technology. Therefore we need the availability of an information technology dictionary. Dictionary of the term information technology that circulates a lot in the form of printed media / books. However, dictionaries in the form of print media have a weakness in the form of a search process that takes a long time and cannot be updated at any time, therefore a digital / electronic dictionary exists to overcome this weakness. The dictionary application of the term information technology by applying the Boyer Moore algorithm in the search for the term word can help it. The Boyer Moore algorithm applies the good suffix principle where the character being searched for is parallel to the like character, as well as the principle of a bad character where if the character does not have similarities, it is immediately eliminated. The field used in this search is the word term, by entering the term search term as input in the search column, then we can find out the meaning of the term word. The results of this application display the entire pattern that matches the text, with a total search time below 300 milliseconds.

ARTICLE INFO

Article History:
Submitted/Received 20 Jun 2022
First Revised 14 Aug 2022
Accepted 09 Aug 2022
First Available online 22 Sep 2022
Publication Date 01 Oct 2022

Keyword:
Boyer Moore Algorithm, Information Technology, Dictionary.

1. INTRODUCTION

Community needs for IT-based technology services are very varied, one of which is the need for dictionaries, dictionaries have various types ranging from general dictionaries such as language dictionaries to dictionaries of special terms such as political dictionaries, economic dictionaries, language dictionaries and so on (Badruzaman, 2019).

The emergence of many new terms in information technology because of the rapid development of technology makes people who pursue the field of technology must know and learn things related to information technology (Mulyadi, et al., 2019). Constraints that the author found from observation and interviews with Mr Nugraha at PT. Erefka Tiga Pilar Utama, there are still a number of apprentice employees and students who are not very familiar with terms in information technology. Based on this, a dictionary of information technology terms in the form of printed media/books was made. However, dictionaries in the form of print media have weaknesses in the form of a search process that takes a long time, therefore a digital/electronic dictionary exists to overcome these weaknesses.

The dictionary adaptation of the term information technology from printed forms to digital/electronic media, especially the web, is still very limited in terms of quantity (Ivić, I. (2019). From the search results using search engines on the internet there are only a few services including the website www.pnri.go.id/Istilah-Kom-puter.aspx. This website also uses conventional HTML technology that requires users to reload all pages to get the information they need.

Dictionary of information technology is a support that can help someone to understand important terms in information technology (Husaini, 2017). It contains important terms to know. Therefore, it is very useful if the dictionary of the term information technology is owned by everyone, especially those who work in the IT field. However, when viewed from the terms of information technology is very much and requires a lot of time so that the process of finding words in the dictionary is getting slower and inefficient.

To overcome these problems, an application dictionary for information technology is needed that can simplify and shorten the time in its use. To speed up and simplify the search process, an algorithm is needed that can maximize search process time. Algorithms are logical steps in solving a problem systematically arranged (Gad, A. G. 2022). The search algorithm is currently growing, there are algorithms that process matching strings from right to left and from left to right. Search for the meaning of a word (string) is something that is very much needed in a dictionary, especially dictionaries of the term information technology, thus using a string search algorithm as a solution to overcome this.

The following is a literature review that has been carried out:

1.1. Dictionary

Dictionary is a type of reference book that describes a word that contains meaning. Serves to help get to know new words. In addition to explaining a word, the dictionary also has a guideline or designation, the origin (etymology) of something said and also an example of use for something said (Anton, 2016).

The equivalent of dictionary words in English is a dictionary, began to be used in writing in 1526 and comes from a Latin word, namely dictionarum. This word is derived from the word dictio which means word or say. The equivalent in Dutch is woordenboek which is

DOI: https://doi.org/10.17509/integrated.v4i2.xx
p- ISSN 2656-9760 e- ISSN 2657-0769

distinguished from woordenschat which in Indonesian is matched with vocabulary or vocabulary (Saptiani, 2017).

1.2. Boyer Moore's Algorithm

The Boyer Moore algorithm is one of the string search algorithms, published by Robert S. Boyer, and J. Strother Moore in 1977. This algorithm is considered to be the most efficient algorithm in general applications. Unlike the string search algorithm found earlier, the Boyer Moore algorithm starts matching characters from the right of the pattern. The idea behind this algorithm is that by starting matching characters from the right, and not from the left, more information will be obtained. The Boyer Moore algorithm includes the most efficient string matching algorithm compared to other string matching algorithms. Because of its efficient nature, many string matching algorithms have been developed by relying on the Boyer Moore algorithm concept (Darmawan, *et al*., 2018).

1.3. How the Boyer Moore algorithm works

1.3.1. How to calculate heuristic occurrence tables:

Example pattern: MOORE

Character length: 5

The Calculate Heuristic Occurrence (Step 1) can be seen in the following **Table 1.**

Table 1. Calculate Heuristic Occurrence (Step 1).

Kriteria	**0**	**1**	**2**	**3**	**4**	**5**
Pattern	M	O	O	R	E	
Occurrence Heuristic						

The steps for giving the value are as follows:

Perform calculations, OH = (length-1-index)

length = character length = 5

The first character is "M" with Index = 0

OH = (5 - 1 - 0 = 4) then the character value "M" = 4

The second character is "O" with index = 1

OH = (5 - 1 - 1 = 3) then the character value "O" = 3

The third character is "O" with index = 2

OH = (5 - 1 - 2 = 2) then the character value "O" = 2

The fourth character is "R" with index = 3

OH = (5 - 1 - 3 = 1) then the character value "R" = 1

DOI: http://dx.doi.org/10.17509/v4i2.xx
p- ISSN 2775-6793 e- ISSN 2775-6815

The fifth character is "E" with index = 4

OH = (5 - 1 - 4 = 0) then the character value "E" = 0

The Calculate Heuristic Occurrence (Step 2) can be seen in the following **Table 2.**

Table 2. Calculate Heuristic Occurrence (Step 2).

Kriteria	**0**	**1**	**2**	**3**	**4**	**5**
Pattern	M	O	O	R	E	
Occurrence Heuristic	4	3	2	1	0	

If there is a recurring character take the smallest OH value, in this case there is a character "O" that is worth 3 and 2, then make the character "O" worth 2.

The Calculate Heuristic Occurrence (Step 3) can be seen in the following **Table 3**.

Table 3. Calculate Heuristic Occurrence (Step 3).

Kriteria	**0**	**1**	**2**	**3**	**4**	**5**
Pattern	M	O	O	R	E	
Occurrence Heuristic	4	2	2	1	0	

1.3.2. How to calculate the match heuristic table

Example pattern: MOORE

Character length: 5

The Calculate the match Heuristic (Step 1) can be seen in the following **Table 4.**

Table 4. Calculate the match Heuristic (Step 1).

Kriteria	**0**	**1**	**2**	**3**	**4**	**5**
Pattern	M	O	O	R	E	
Match Heuristic						

The steps for giving the value are as follows:

1. Give the value 1 (default MH value) at the largest index, in this case the biggest index is "E".
2. Compare the second character, if the character "E" has been found / has been matched, but the character before "E" is not "R", then check whether the matching character in the text

DOI: https://doi.org/10.17509/integrated.v4i2.xx
p- ISSN 2656-9760 e- ISSN 2657-0769

is in the next pattern. , and give the displacement value to the MH table, in this case "E" is suitable.

The Calculate the match Heuristic (Step 2) can be seen in the following **Table 5.**

Table 5. Calculate the match Heuristic (Step 2).

Kriteria	**0**	**1**	**2**	**3**	**4**	**5**
Text	M	O	O	**L**	E	
Pattern	M	O	O	**R**	E	
Match Heuristic						

3. Continue the second step for the next character until the end of the pattern, the pattern mismatch then compare the characters that have matched the characters that have not been compared to the pattern, if there is, align the characters, otherwise the shift will be as long as the character, and give the shift values to table MH.

The Calculate the match Heuristic (Step 3) can be seen in the following **Table 6.**

Table 6. Calculate the match Heuristic (Step 3).

Kriteria	**0**	**1**	**2**	**3**	**4**	**5**
Text	M	O	O	**L**	E	
Pattern	M	O	O	**R**	E	
Match Heuristic	5	5	5	5	1	

Example of a search with the Boyer Moore algorithm:

If the characters that are compared are not in the pattern then the value of the shift is along the number of patterns.

To shift the OH and MH values, take the biggest value as a shift decision.

If all characters match, the pattern has been found in the text.

Examples of Value Occurrence Heuristics and Match Heuristics

The Calculate the match Heuristic (Step 4) can be seen in the following **Table 7.**

Table 7. Calculate the match Heuristic (Step 4).

Kriteria	**0**	**1**	**2**	**3**	**4**	**5**
Pattern	M	O	O	R	E	

DOI: http://dx.doi.org/10.17509/v4i2.xx
p- ISSN 2775-6793 e- ISSN 2775-6815

Occurence Heuristic	4	2	2	1	0
Match Heuristic	5	5	5	5	1

Example of a search implementation in the first process text

The Calculate the match Heuristic (Step 5) can be seen in the following **Table 8.**

Table 8. Calculate the match Heuristic (Step 5).

Text	M	O	O	L	I	M	O	O	R	E
Pattern	M	O	O	R	E					
Index	0	1	2	3	4					

OH = I (Text) = 5

MH = index [4] (Pattern) = 1

5> 1, so the pattern is shifted 5 characters

The Calculate the match Heuristic (Step 6) can be seen in the following **Table 9.**

Table 9. Calculate the match Heuristic (Step 6).

Text	M	O	O	L	I	**M**	**O**	**O**	**R**	**E**
Pattern						**M**	**O**	**O**	**R**	**E**
Index						0	1	2	3	4

All characters match, meaning the pattern has been found in the text.

1.4. Rapid Application Development Method (RAD)

According to Widiyanto (2018), is a model of the development of linear sequential software processes that emphasize very short development cycles. This RAD model is a "high-speed" adaptation of a linear sequential model in which rapid development is achieved using a component-based construction approach if needs are well understood, the RAD process allows the development team to create a "complete functional system" in a relatively short period of time (approximately 60-90 days).

1.5. Phase in Rapid Application Development (RAD)

There are four phases in RAD that involve analysts and users in the stages of needs planning, design, development, and implementation (Shelly, *et al.,* 2012).

1.5.1. Phase Requirement Planning

The need planning stage combines elements of the system planning and system analysis. Users, managers, and IT staff members discuss and agree on business requirements, project

DOI: https://doi.org/10.17509/integrated.v4i2.xx
p- ISSN 2656-9760 e- ISSN 2657-0769

scope, constraints, and system requirements (Tritularsih, & Sutopo, 2017). The need planning phase ends when the team agrees on the issues that exist and obtains authorization management (Supriyatna, 2016).

1.5.2. Workshop Design Phase

During the workshop design phase, users interact with system analysts and develop models and prototypes that represent all input, process and output systems. RAD groups or subgroups usually use a combination of JAD and CASE tools to translate user requirements into work models. User design is a continuous, interactive process that allows users to understand, modify, and finally approve a working model of a system that meets needs.

1.5.3. Construction

The construction phase focuses on program and application development. In RAD, users continue to participate and can still suggest changes or improvements to the appropriate development report.

1.5.4. Implementation

The cutover stage is the final task in the implementation, such as data conversion, testing, changes to the new system, and user training. Compared to traditional methods, the entire process is compressed. As a result, new systems are built, delivered, and placed in faster operations.

2. METHODS

2.1. Method of Collecting Data

2.1.1. Field Study

2.1.1.1. Observation

The essence of observation is the presence of visible behavior and goals to be achieved. Behavior that appears to be a behavior that can be seen directly by the eye, can be heard, can be calculated, and can be measured [6]. In this study, the author made observations at PT. Erefka Three Main Pillars on October 20, 2018 and November 10, 2018.

2.1.1.2. Interview

According to Gorden, quoted by Haris Herdiansyah in his book entitled "Qualitative Research Methodology" the definition of interview is an interview is a conversation between two people, one of which aims to explore and obtain information for a particular purpose. In the research conducted by the author at PT. Erefka Tiga Pilar Utama, which became the author of the author to obtain data, namely Mr. Nugraha as the owner of PT. Erefka Tiga Pilar Utama, the interview was conducted on November 20, 2018.

2.1.2. Library Studies

2.1.2.1. Book Reference

Some books or literature used by the author in designing and compiling this research report include:

1. Rekayasa Perangkat Lunak (Rosa, *et al*., 2011).

DOI: http://dx.doi.org/10.17509/v4i2.xx
p- ISSN 2775-6793 e- ISSN 2775-6815

2. Membangun Website Dinamis Interaktif dengan PHP-MySQL (Windows & Linux) Untuk Tingkat Pemula dan Tingkat Lanjut (Sukarno & Mohamad., 2006).

3. Mastering Ajax dan PHP (Kadir & Abdul., 2009).

2.1.2.2. Similar Literature

In the study of similar literature, the authors collected data from five similar journals related to this study, in gathering information about almost the same problem, a comparison was needed in this regard so that the system could be better.

2.1.2.3. Online Data Search

At this stage the author conducts the information search process by studying each concept related to the Boyer Moore method and its implementation in other case studies, UML (Unified Modeling Language), and RAD (Rapid Application Development). Online search is also used for solving problems related to the source code in making dictionary applications the term information technology.

2.2. System Development Method

The system development method that I use to develop this system is the RAD (Rapid Application Development) development method using the theory of Gary et al (2011). While the tools used are UML (Unified Modeling Language) notation using the Salahuddin theory.

3. RESULTS AND DISCUSSION

3.1. Design of Boyer Moore Algorithm

The following is a description of the steps of the Boyer Moore algorithm in dictionary searches that occur (Ojugo & Oyemade, 2021) when a user enters the word "list".

Text (Y) : Mailing list

Pattern (X) : list

Bc : Bad Character

The Design of Boyer Moore Algorithm (Step 1) can be seen in the following **Table 10.**

Table 10. Design of Boyer Moore Algorithm (Step 1).

Index	0	1	2	3	4	5	6	7	8	9	10	11
Y	M	a	i	I	i	N	g		1	i	s	t
X	1	i	s	t								
Bc	3	2	1	4								

In the first step character adjustment will begin in the pattern "t" character where the character "t" is not the same as the character above in the text (miss match) that is "l", then the pattern will shift a number of bad characters "l" to the text that is 3 steps.

DOI: https://doi.org/10.17509/integrated.v4i2.xx
p- ISSN 2656-9760 e- ISSN 2657-0769

The Design of Boyer Moore Algorithm (Step 2) can be seen in the following **Table 11**.

Table 11. Design of Boyer Moore Algorithm (Step 2).

Index	0	1	2	3	4	5	6	7	8	9	10	11
Y	M	a	i	I	i	N	g		1	i	s	t
X				1	i	s	t					
Bc				3	2	1	4					

In the second step the adjustment will begin again at the character "t" the rightmost character pattern where the character "t" is not the same as the character above in the text "g", then the pattern will shift a number of bad characters "g" to the text. Because "g" is not available in a bad character pattern, what is taken is the number of patterns that is 4 characters, it will shift by 4 steps.

The Design of Boyer Moore Algorithm (Step 3) can be seen in the following **Table 12.**

Table 11. Design of Boyer Moore Algorithm (Step 3).

Index	0	1	2	3	4	5	6	7	8	9	10	11
Y	M	a	i	I	i	N	g		1	i	s	t
X									1	i	s	t
Bc									3	2	1	4

In the fourth step, the adjustment will begin again on the character "t" the rightmost character pattern, where the character "t" matches the character in the text, as well as the characters "s", "i", and "l" in the pattern. so that the pattern has been found in the text in the dictionary application database the term information technology and will issue results in terms of words that have been found in the search, namely the word list.

3.2. Pseudocode of Boyer Moore's Algorithm

The following is a part of the boyer moore algorithm pseudocode on the search function used by the author in making dictionary applications the term web-based information technology.

DOI: http://dx.doi.org/10.17509/v4i2.xx
p- ISSN 2775-6793 e- ISSN 2775-6815

```
procedure BoyerMooreSearch(
    input m, n : integer,
    input P : array[0..n-1] of char,
    input T : array[0..m-1] of char,
    output ketemu : array[0..m-1] of
   boolean
)
Deklarasi:
i, j, shift, bmBcShift, bmGsShift:
integer
BmBc : array[0..255] of integer
BmGs : array[0..n-1] of integer
Algoritma:
preBmBc(n, P, BmBc)
preBmGs(n, P, BmGs)
i:=0
while (i<= m-n) do
j:=n-1
while (j >=0 n and T[i+j] = P[j]) do
j:=j-1
endwhile
```

3.3.

System Testing

The following is a system testing process in the dictionary application, the term web-based information technology that the author uses using the black box method, consisting of several columns, including a description column containing an explanation of the user's activities on the system desired, the actual results column contains an explanation of the actual results on the system after testing.

The System testing can be seen in the following **Table 14.**

DOI: https://doi.org/10.17509/integrated.v4i2.xx
p- ISSN 2656-9760 e- ISSN 2657-0769

Table 14. System Testing.

No	Description	Expected Results	Actual Results
1.	*Admin Login*	*Admin* can *login* application	Appropriate
2.	*Create*, *Read*, *Update*, *Delete* Dictionary Data	*Admin* can *create, read, update, delete* dictionary data in the application	Appropriate
3.	*Admin* prints dictionary data	*Admin* can print dictionary data in the application	Appropriate
4.	*Admin Import* Dictionary Data	*Admin* can *import* data in *excel format* in the application	Appropriate
5	*Admin Create*, *Read*, *Update*, *Delete* Data Category	*Admin* can *create, read, update, delete* data *category* application	Appropriate
6.	*Users* Search for Data Dictionary	*Users* can search data in the application	Appropriate
7.	*Users* Send Messages	*Users* can *input* messages in the application	Appropriate

3.4. Performance of the Boyer Moore Algorithm

To find out the performance of the Boyer Moore algorithm in this study, the author utilizes the features of the browser contained in the network panel of the developer tools to find out the time obtained in the term search for information technology. Boyer Moore's algorithmic performance is taken based on three times, namely:

Request Sent/Sending, which is the process of sending data that is input to the client for the server.

Waiting (Time To First Byte), knowing the length of time from the sending process to the process of retrieving data from the server.

Downloading is the process of retrieving data from server to client.

Testing is done by inputting data in the search column with several characters as follows:

The Performance of the Boyer Moore Algorithm can be seen in the following **Table 15**.

DOI: http://dx.doi.org/10.17509/v4i2.xx
p- ISSN 2775-6793 e- ISSN 2775-6815

Table 15. Performance of the Boyer Moore Algorithm.

No	Teks yang diketik	Time (ms)	Timeline Request/Response (Millisecond)		
			Sending	Waiting (TTFB)	Downloading
1	Bi	238	0.16	213.38	7.17
2	Bin	239	0.14	229.69	6.99
3	Bina	239	0.15	210.09	7.37
4	Binar	251	0.22	229.91	5.75
5	Binary	256	0.16	225.85	9.19

The following is a further test of the performance of the Boyer Moore algorithm in the dictionary application of the term information technology, testing is done by classifying some data from the dictionary database the term information technology as follows.

The Text with the word Antivirus can be seen in the following **Table 16**.

Table 16. Text with the word Antivirus

Teks yang diketik	Data Pada Database	Time (ms)	Timeline Request/Response (Millisecond)		
			Sending	Waiting (TTFB)	Downloading
Antivirus	70	228	0.16	212.72	8.74
	140	230	0.16	216.06	8.91
	200	265	0.18	242.27	9.56

The Text with the word Binary can be seen in the following **Table 17.**

Table 17. Text with the word Binary.

Teks yang diketik	Data Pada Database	Time (ms)	Timeline Request/Response (Millisecond)		
			Sending	Waiting (TTFB)	Downloading

DOI: https://doi.org/10.17509/integrated.v4i2.xx
p- ISSN 2656-9760 e- ISSN 2657-0769

Binary	70	224	0.16	209.32	7.81
	140	229	0.17	214.73	8.33
	200	240	0.19	229.23	9.21

The Text with the word Hashtag (#) can be seen in the following **Table 18.**

Table 17. Text with the word Binary.

Teks yang diketik	**Data Pada Database**	**Time (ms)**	**Timeline Request/Response (Millisecond)**		
			Sending	**Waiting (TTFB)**	**Downloading**
Hashtag (#)	70	223	0.15	208.75	7.62
	140	227	0.17	215.47	8.57
	200	236	0.18	228.16	9.83

From the results of the above tests it can be concluded that:

1. The difference in the amount of data in the database does not significantly affect the performance of the Boyer Moore algorithm, it is evident that this algorithm can produce word search delivery times below 0.30 milliseconds, waiting times below 250 milliseconds, and download times or download data below 10 milliseconds, with a total search time below 300 milliseconds.
2. In doing a word search, this application successfully provides information search results well.
3. Word search in dictionary applications the term information technology supports search by using conjunctions or symbols as input to the search term words as long as the word is available on the application database.
4. The performance of the Boyer Moore algorithm that is applied to the application is also influenced by several factors, including the amount of data in the database, writing the source code in the application, the state of network traffic when searching for words and performance factors on the hardware used.

3.5. Application Design

The following is a display of the application that shows the process of managing dictionary data by the admin. The application design can be seen in the following **Figure 1, Figure 2, Figure 3, Figure 4,** and **Figure 5**.

DOI: http://dx.doi.org/10.17509/v4i2.xx
p- ISSN 2775-6793 e- ISSN 2775-6815

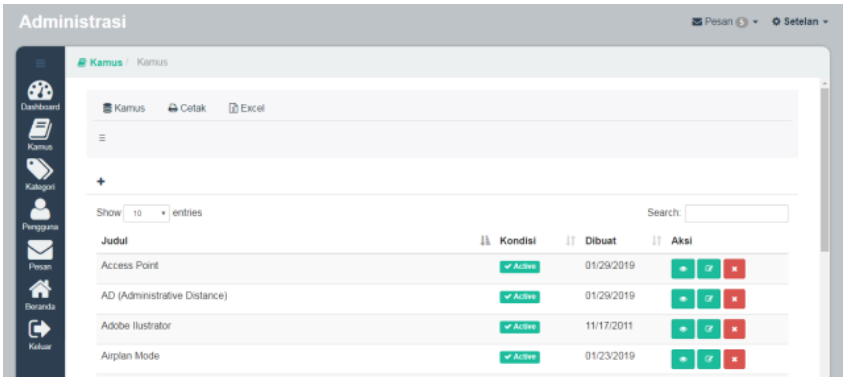

Figure 1. Page view data dictionary.

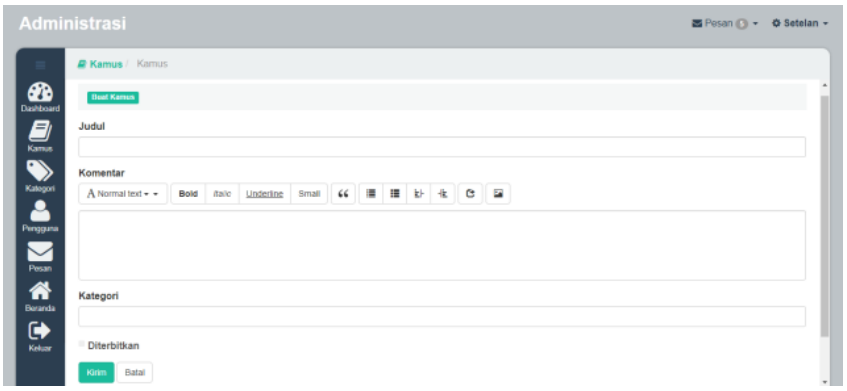

Figure 2. Dictionary data add page.

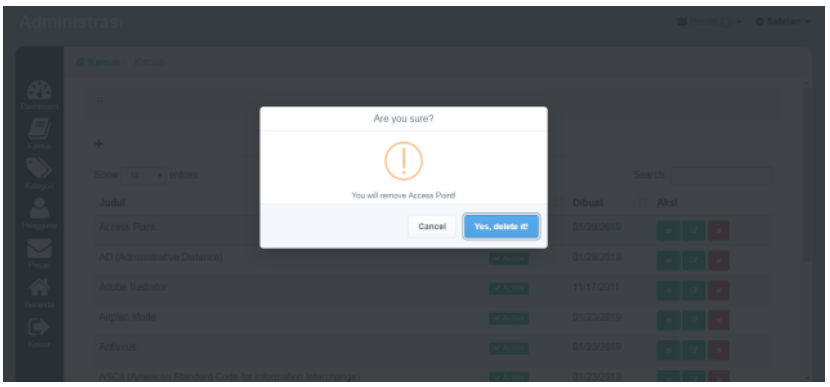

Figure 3. The page deletes dictionary data.

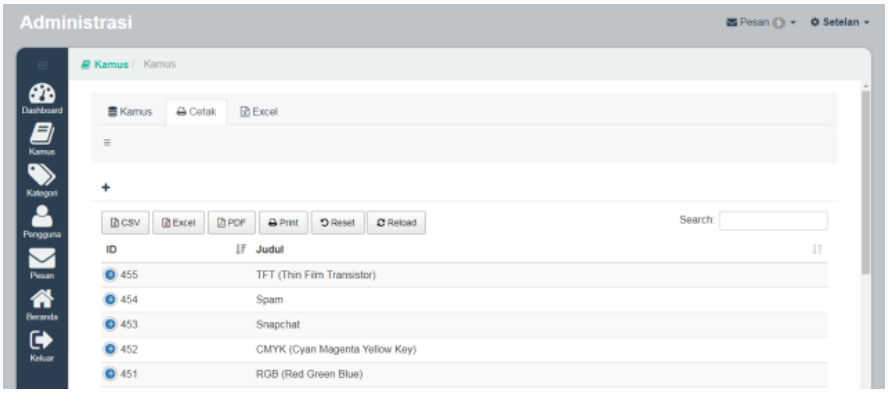

Figure 4. Dictionary data print page.

DOI: https://doi.org/10.17509/integrated.v4i2.xx
p- ISSN 2656-9760 e- ISSN 2657-0769

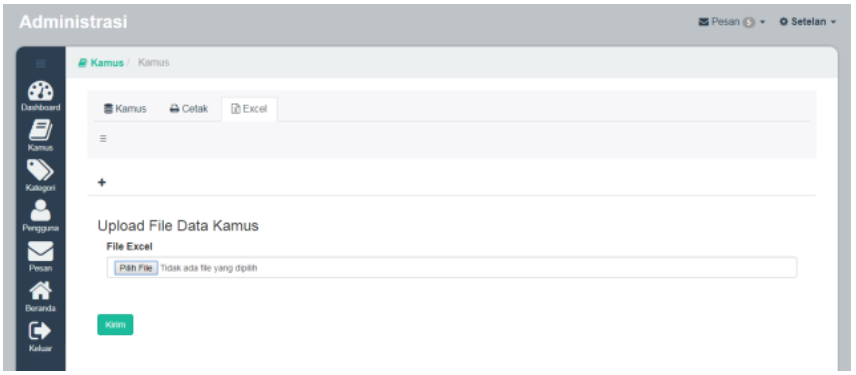

Figure 5. The dictionary data import page.

4. CONCLUSION

Based on research conducted with several stages of word search testing using the Boyer Moore method, that the application of the Boyer Moore method in the dictionary application the term information technology was successfully carried out, can be seen from several word input and produce the appropriate output. In inputting words in the text field search column, using the word "binary" as a search pattern, the lowest total time span of 238 milliseconds is obtained by input using 2 characters (B, i), then 256 milliseconds are obtained by input using 6 characters (B, i, n, a, r, y), until the highest time obtained 265 milliseconds by input using 8 characters (A, n, t, i, v, i, r, u, s), searches are processed into the boyer moore algorithm and successfully performed. In testing conducted by classifying some data from the database based on the amount of data in the dictionary database the term information technology, it was found that the difference in the amount of data in the database slightly affected the performance of the boyer moore algorithm applied to the application.

4.1. Suggestion

No system is truly perfect, but to achieve this, a system must be developed to improve performance to cover the system's shortcomings. Therefore, there are some that need to be considered to improve the dictionary application for the term information technology to be better, including:

1. In further development, it is expected that similar applications based on mobile operating systems such as Android and IOS, which data from this application can be synchronized into it.

2. It is expected that in the next development there will be a real time feature in the dictionary search word application term information technology, so that when the word you want to search is entered, search results will appear without clicking on the search button.

3. In further development it is expected that users can contribute to being able to add vocabulary that is not yet available in the application with the approval of the admin.

4. In terms of application security must be improved.

5. It is expected that in further development users and experts in information technology can communicate through applications both on the web and smartphone.

6. Vocabulary in the dictionary must be further reproduced.

DOI: http://dx.doi.org/10.17509/v4i2.xx
p- ISSN 2775-6793 e- ISSN 2775-6815

5. REFERENCES

Anton Suryoprayogo, S. (2016). Implementasi algoritma brute force pada perancangan aplikasi kamus bahasa sunda berbasis android. *Jurnal Informatika dan Bisnis*, 5(1), 1-12.

Badruzaman, D. (2019). Kajian hukum tentang internet mobile dalam upaya pencegahan dampak negatif teknologi informasi dan komunikasi di Indonesia. *Ajudikasi: Jurnal Ilmu Hukum,* 3(2), 135-152.

Darmawan, R. I., Setianingrum, A. H., & Arini, A. (2018). Implementasi algoritma Boyer Moore pada aplikasi kamus istilah kebidanan berbasis web. *Query: Journal of Information Systems,* 2(1), 53-62.

Gad, A. G. (2022). Particle swarm optimization algorithm and its applications: a systematic review. *Archives of computational methods in engineering*, *29*(5), 2531-2561.

Husaini, M. (2017). Pemanfaatan teknologi informasi dalam bidang pendidikan (e-education). *MIKROTIK: Jurnal Manajemen Informatika,* 2(1).

Ivić, I. (2019). Printed and digital media: Printed and digital textbooks. *Center for Educational Policy Studies Journal*, *9*(3), 25-49.

Kasenda, L. M., & Sentinuwo, S. (2016). Sistem monitoring kognitif, afektif dan psikomotorik siswa berbasis android. *Jurnal Teknik Informatika,* 9(1), 1-9.

Mulyadi, M., Zulkarnain, I., & Laugu, N. (2019). Adaptasi pustakawan dalam menghadapi kemajuan teknologi. *Berkala Ilmu Perpustakaan Dan Informasi,* 15(2), 163-174.

Ojugo, A. A., & Oyemade, D. A. (2021). Boyer moore string-match framework for a hybrid short message service spam filtering technique. *IAES International Journal of Artificial Intelligence*, *10*(3), 519.

Saptiani, S. (2017). Pengembangan media pembelajaran kamus tematik tiga bahasa (indonesia, arab, inggris) untuk menstimulasi kognitif anak usia dini. *Bunayya: Jurnal Pendidikan Anak,* 1(2), 100-118.

Supriyatna, A. (2016). Sistem informasi pengajuan klaim jaminan pemeliharaan kesehatan masyarakat berbasis web. *Paradigma,* 18(2), 1-10.

Widiyanto, W. W. (2018). Analisa metodologi pengembangan sistem dengan perbandingan model perangkat lunak sistem informasi kepegawaian menggunakan waterfall development model, model prototype, dan model rapid application development (rad). *Jurnal Informa: Jurnal Penelitian dan Pengabdian Masyarakat,* 4(1), 34-40.

DOI: https://doi.org/10.17509/integrated.v4i2.xx
p- ISSN 2656-9760 e- ISSN 2657-0769