

## Efficiency of hybrid algorithm for COVID-19 online screening test based on its symptoms

Mohd Kamir Yusof<sup>1</sup>, Wan Mohd Amir Fazamin Wan Hamzah<sup>1</sup>, Nur Shuhada Md Rusli<sup>2</sup>

<sup>1</sup>Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin, Terengganu, Malaysia

<sup>2</sup>Department of Information and Communication, Politeknik Balik Pulau, Pulau Pinang, Malaysia

### Article Info

#### Article history:

Received Oct 27, 2020

Revised Oct 15, 2021

Accepted Nov 19, 2021

#### Keywords:

AJAX

Coronavirus

COVID-19 symptoms

Hybrid algorithm

JQuery

JSON

### ABSTRACT

The coronavirus COVID-19 is affecting 196 countries and territories around the world. The number of deaths keep on increasing each day because of COVID-19. According to World Health Organization (WHO), infected COVID-19 is slightly increasing day by day and now reach to 570,000. WHO is prefer to conduct a screening COVID-19 test via online system. A suitable approach especially in string matching based on symptoms is required to produce fast and accurate result during retrieving process. Currently, four latest approaches in string matching have been implemented in string matching; characters-based algorithm, hashing algorithm, suffix automation algorithm and hybrid algorithm. Meanwhile, extensible markup language (XML), JavaScript object notation (JSON), asynchronous JavaScript XML (AJAX) and JQuery technology has been used widely for data transmission, data storage and data retrieval. This paper proposes a combination of algorithm among hybrid, JSON and JQuery in order to produce a fast and accurate results during COVID-19 screening process. A few experiments have been by comparison performance in term of execution time and memory usage using five different collections of datasets. Based on the experiments, the results show hybrid produce better performance compared to JSON and JQuery. Online screening COVID-19 is hopefully can reduce the number of effected and deaths because of COVID.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



### Corresponding Author:

Nur Shuhada Md Rusli

Department of Information and Communication, Politeknik Balik Pulau

11000 Balik Pulau, Pulau Pinang, Malaysia

Email: shuhadarusli94@gmail.com

## 1. INTRODUCTION

COVID-19 is a virus called SARS-CoV-2. This virus is appeared to have first emerged in Wuhan, China [1]. SARS-CoV-2 is considered a novel human-infecting Beta coronavirus [2]. SARS-CoV-2 is a member of the family Coronaviridae and order Nidovirales [3]. This family consist of two subfamilies: Coronavirinae and Torvirinae. Subfamily of Coronavirinae is divided into four genera: (i) Alphacoronavirus which is contains the human coronavirus 229E (HCoV-229E) and human coronavirus NL63 (HCoV-NL63); (ii) Betacoronavirus includes human coronavirus OC43 (HCoV-OC43), severa acute respiratory syndrome human coronavirus (SARS-HCoV), human coronavirus HKU1 (HCoV-HKU1), and middle eastern respiratory syndrome coronavirus (MERS-CoV); (iii) Gammacoronavirus includes viruses of whales and birds; and (iv) Deltacoronavirus includes isolated from pigs and birds [4]. SARS-CoV-2 is an enveloped and positive-strand RNA (+ssRNA) virus.

COVID-19 has become a worldwide threat because of death of 26,495 individuals and infected more than 570,000 [5]. A few symptoms such as a fever, dry cough, and tiredness, and in mild cases people

have a potential to get this virus [1]. According to several cases, people with the virus can develop difficulty of breathing, and may ultimately experience organ failure. According to World Health Organization (WHO), most people infected with the COVID-19 is recover without requires a special treatment [6]. However, older people and those with underlying medical problems such as cardiovascular disease, diabetes, chronic respiratory disease, and cancer are more likely to develop serious illness.

The COVID-19 virus is primarily spreading through droplets of saliva or discharge from the nose when an infected person coughs or sneezes. The best way to prevent or slow down transmission of infection COVID-19 is protecting yourself and others by washing your hands or using alcohol-based run frequently and not touching your face [1]. The number of infected COVID-19 is slightly increasing day by day and now reach to 570,000. In order to manage this amount of data, a suitable online system needs to avoid any delay during the execution process. This study was conducted to propose an efficient storing and retrieving data approach to execute a large amount of data at concurrent time access.

## 2. RESEARCH METHOD

Symptoms of COVID-19 are different among individuals because of ranging from asymptomatic infection to severe respiratory failure [7]. According to Italian population study which is conducted by the town of Vo Euganeo by Dr. Lavezzo and colleagues, around 50-70% of individuals with positive polymerase chain reaction (RT-PCR) throat swab results remain symptoms and a further small percentage (about 10% of all symptomatic patients) present dyspnoea, severe interstitial pneumonia, acute respiratory distress syndrome (ARDS) and multi-organ dysfunction. Meanwhile, the vast majority of individuals with symptoms had one or more coexisting medical conditions such as hypertension, diabetes, and cardiovascular disorders with elevated case facilities amongst elderly and frail patients [8], [9]. Others common symptoms of the disease such as fever, cough, fatigue, slight dyspnoea, sore throat, headache, and conjunctivitis. That way, it is difficult to differentiate COVID-19 from other respiratory disease [10]-[12]. Meanwhile, involvement of gastrointestinal such as diarrhoea, nausea and vomiting are lower percentage of cases. Table 1 shows summary of symptoms observed in various cohorts of patient. Based on Table 1, fever and cough contributed high percentages between 44-98%. Meanwhile, the second high percentages between 19-83% are dyspnoea and sputum. Other's symptoms such as sore throat, headache, diarrhoea, nausea/vomiting and myalgia contributed less than 15%.

Table 1. Symptoms observed in various cohorts of patient

	Guan <i>et al.</i> [13]	Chen <i>et al.</i> [14]	Shi <i>et al.</i> [15]	Huang <i>et al.</i> [16]	Yang <i>et al.</i> [17]
Patients	1081	99	21	41	52
Fever	473 (44%)	82 (83%)	18 (86%)	40 (98%)	46 (89%)
Dyspnoea	205 (19%)	31 (31%)	9 (43%)	22 (43%)	33 (64%)
Cough	745 (69%)	81 (82%)	15 (71%)	31 (76%)	40 (77%)
Sputum	370 (34%)	-	3 (14%)	11 (27%)	-
Rhinorrhoea	53 (5%)	4 (4%)	5 (24%)	-	3 (6%)
Sore throat	153 (14%)	5 (5%)	-	-	-
Headache	150 (14%)	8 (8%)	2 (10%)	2 (5%)	3 (6%)
Diarrhoea	42 (4%)	2 (2%)	1 (5%)	1 (2%)	-
Nausea/vomiting	55 (5%)	1 (1%)	2 (10%)	-	2 (4%)
Myalgia	164 (15%)	11 (11%)	-	-	6 (12%)

## 3. MATCHING ALGORITHM, AJAX AND JQUERY TECHNOLOGY

The motivation of this study is to develop an application for online screening process COVID-19 based on a suitable architecture. The architecture must be able to manage multiple users at concurrent time access based on symptoms in Table 1. Increasing number of infected COVID-19 required fast data transmission between client and server. In this paper, four different approaches for string matching have been identified: i) character-based approach, ii) hashing based approach, and iii) suffix automata-based approach. Meanwhile, AJAX and jQuery are the most relevant technologies for fast data transmission between client and server side.

One of the traditional approaches in string matching problems is character-based approach [18]. In this approach, each character is compared from left to right individually at the cost of extra time requirements. Two stages involved in this approach: searching and shift phases. Based on the experiments, results in the corpus of Arabic, English, Chinese, Italian, and French texts are promising. However, the time complexity of the algorithm may increase when searching for multiple patterns in given text.

Another approach for string matching is hashing [19]. Hashing value has been used for matching process and conducts comparison from left to right. For instance, a given pattern  $p$  with characters  $m$ , the

hashing values are calculated first. Then, the hash value is used for matching between a given text  $t$  and a pattern  $p$ . After the hash function is used, the searching phase uses character comparison. Therefore, the algorithm involves two key steps: pre-processing and searching process. In pre-processing phase, a string of  $c$  characters is converted into a string of  $d$  decimal number. The pattern  $p$  is divided by pre-defined prime number  $q$ . Modulus operation is the used to calculate the remainder of pattern  $p$  with  $q$ . For each shift that ranges from shift ( $s = 0$  to  $n - m$ ), the remainders of pattern and are compared for matching. Once a match is found, brute-force approach is implemented to verify the result of matching. This approach saves large amount of computation as it compares integer values instead of characters. However, the limitation of this approach is hash collision. Hash collision is occurring when two different strings map to the same number.

Suffix automation is an automation that compares two related but distinct automata constructors; deterministic acyclic finite state automation and suffix automation for matching [20]. Based on this approach, it can be defined as  $D(p) = \{Q, q_0, F, \Sigma, \delta\}$ . Here,  $Q = \{q_1, q_2, q_3, q_m\}$  is set of states,  $F = \{q_m\}$  is set to accept states, and  $\delta = Q \times \Sigma \rightarrow Q$  is the transition function. This approach uses a directed acyclic graph in which nodes/vertices are called states, and edges between nodes are considered a transition between the states. One of the states (node) denoted by  $q_0$  is called the initial state of the suffix automation from where we can reach to all other states in the automation. For instance, we have a pattern input “abbabb”, each state  $q_0$  denoted by 0 in the example. The suffix automation while traversing from state 0 to terminal nodes must represent a suffix that is case, the possible suffixes of “abbabb” are “b”, “bb”, “abb”, “babb”, and “bbabb”. The basic idea in this approach is the text  $t$  is scanned from left to right, and the algorithm decides the number of pattern  $p$  to be shifted to avoid redundant comparisons during a mismatch.

Hybrid algorithm is introduced based on combination the advantages of different algorithms and is better than individual algorithms [21]-[22]. In this approach, one or more characters are combined and placed under automation-based approach. This hybrid algorithm is a modified version of Horspool algorithm [23] with additional string-matching conditions for scanning and matching the text (from left to right) and string pattern (from right to left). This algorithm has two steps. In the first step, the pivoting character in the pattern is searched by computing the character distance and its maximum safe shift. In the second step, the pivot character will be compared to characters of the text. If the pivoting character matches with the characters in the text, then the algorithm starts matching the pattern with text from the right most character until the end of the text. If the matching fails, then hybrid Horspool shift is used for matching. This algorithm is proven achieves good performance for pattern matching on human proteins, text of natural language (e.g., Arabic, Italian, French and Chinese), and E. coli genome.

Asynchronous JavaScript XML (AJAX) is a bundle technology used to simplify the implementation of rich and dynamic web application [24]. AJAX provides advanced and sophisticated user interactions between application/system and users. In fact, the synchronous request-response protocol used by traditional application/system introduces delay between one page to the next one. Features of document object model (DOM) increase the power of AJAX functionality.

JQuery technology is lightweight JavaScript library that emphasises interaction between JavaScript and hypertext markup language (HTML) [25]. JQuery work is almost like AJAX, whose main characteristic is to load data on a web page without reloading the entire page [26]. The advantages of JQuery and AJAX include bandwidth usage and separation of data, format, style, and function [25].

Extensible markup language (XML) and JavaScript object notation (JSON) are the most common data model used by AJAX and JQuery. XML has widely been accepted as the relevant standardisation for representing and exchange the data on the web [27]-[29]. Meanwhile, JSON is alternative data model compared to XML due to its relative simplicity and compactness [30]. JSON provides significant performance over XML, which require extra libraries to retrieve data from document object model (DOM) objects. JSON is estimated to parse up to one hundred times faster than XML in modern browser [29]. This study uses matching algorithms approach in order to improve a performance of data transmission between client and server side. Main task of matching algorithm is to calculate possibility of infected COVID-19 based on each symptom weightiest.

Based on the explanation above, hybrid algorithm is choosen and will be integrated with AJAX and JQuery for searching and retrieving process. There are calculate weightiest for each symptom, choose a suitable technology for data transmission between client side and server side, generate a data model and analyse a performance of the model. Next section describes detail about our proposed method.

#### 4. METHODOLOGY

In our methodology, the first step is to produce a weightiest for each symptom based on Table 1. Table 2 represent value of weightiest for each symptom. Each weightiest are calculated based on (1).

$$\text{Average} = \frac{\text{Total percentage}}{\text{Number of symptoms}} \tag{1}$$

Table 2. Average for each symptom

Symptoms	Guan <i>et al.</i> [13]	Chen <i>et al.</i> [14]	Shi <i>et al.</i> [15]	Huang <i>et al.</i> [16]	Yang <i>et al.</i> [17]	Average
Fever	0.44	0.83	0.86	0.86	0.89	0.776
Cough	0.69	0.82	0.71	0.71	0.77	0.740
Dyspnoea	0.19	0.31	0.43	0.43	0.64	0.400
Diarrhoea	0.04	0.02	0.05	0.50	0.00	0.305
Rhinorrhoea	0.05	0.04	0.24	0.24	0.06	0.126
Headache	0.14	0.08	0.10	0.10	0.06	0.096
Myalgia	0.15	0.11	0.00	0.00	0.12	0.076
Nausea/vomiting	0.05	0.01	0.10	0.10	0.04	0.060
Sore throat	0.14	0.05	0.00	0.00	0.00	0.038
					Total	2.757

The weightiest for each symptom will be used to calculate percentage possibility based on input from users. The second step is to design a suitable architecture for data transmission between client side and server side. Figure 1 shows two different architectures will be implemented in the data transmission between client side and server side. Three main elements involve in this architecture: client-side engine, data model and matching algorithm.

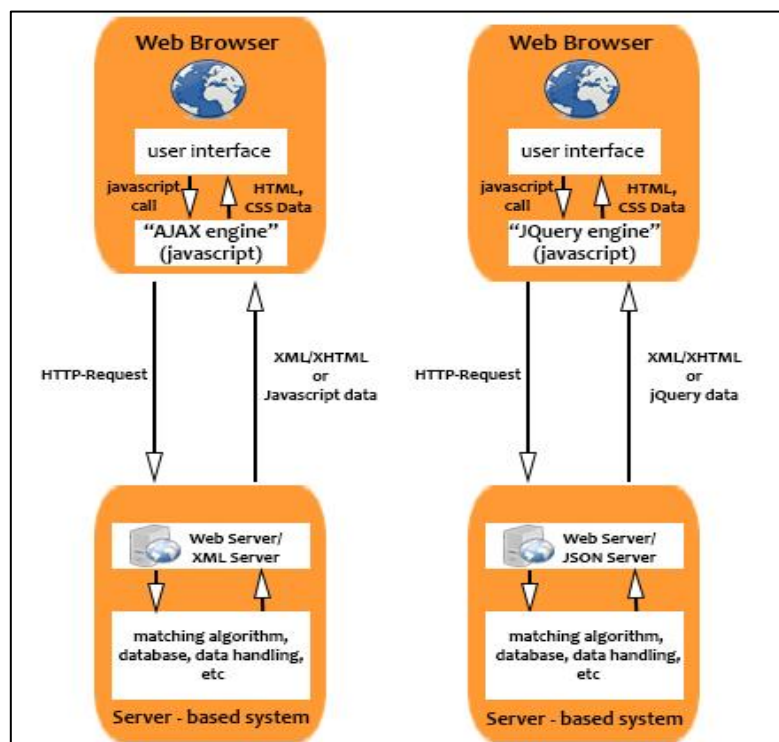


Figure 1. XML and JSON architecture

#### 4.1. Client-side engine

Hybrid algorithm, AJAX and JQuery will be used for client-side engine. Figures 2, 3, and 4 represent a syntax of hybrid algorithm, AJAX and JQuery.

a) Hybrid algorithm

This algorithm will be used to search, and match based on symptoms according to Table 1.

b) AJAX

AJAX is combining the techniques on cascading style sheets (CSS), JavaScript extensible hypertext markup language (XHTML), and document object model (DOM). There are three main elements in AJAX; initial declaration, target uniform resource locator (URL) and execute URL.

c) JQuery

JQuery events are a flexible and simple way of using JavaScript events. JQuery’s application programming interfaces (APIs) provide both wrapper method for common JavaScript events, as well as more detailed application programming interfaces (APIs) in the on (), off () and trigger () methods. Three main components in JQuery; URL or target destination, method type both post/get, and data.

```

Start
Assign pattern p
Find length of pattern, p equal to bi
    If mismatched (where 0 ≤ i < k and bi ≠ bi+1)
        shifted to exactly (k - i + 1) or (k - j + 1) positions
        set position (0 ≤ k < m or j)
    Else
        Repeat the searching process
End
    
```

Figure 2. Hybrid algorithm

```

function fx () {
// initial declaration variable
x = new XMLHttpRequest();
// target URL
http.open(y,z,true);
// execute URL
http.send();
}
    
```

Figure 3. AJAX syntax

```

S(function() {
    S("#form").submit(function(e)
        e.preventDefault();
        S.ajax({
            url: x,
            type: y,
            data: S("#form").serialize(),
            success: function(data) {
                // message success/fail
            }
        });
    });
});
    
```

Figure 4. JQuery syntax

4.2. Data model

This section explained two different data model: XML and JSON. Figures 5 and 6 shows a standard data model structure for XML and JSON.

a) XML

XML is designed as an all-purpose format for organising data. In many cases, we are decided to use XML to store data in a standardised way, rather than creating a new format convention. Figure 5 shows the standard format for XML document. Three main elements in Figure 1, there are root, child and subchild. Roots represent name of object. Child represent name of attribute and subchild represent name of the sub-attribute.

Definition 1: Let, XML file, B = {R, C, H}, where S is represented as object, C is represented attribute name, and H is represented as sub-attributes name.

```

<root>
<child>
<subchild>.....</subchild>
</child>
</root>
    
```

Figure 5. XML schema

b) JSON

JSON file has a standard format or schema. Figure 5 shows the standard of JSON file. Three main elements involved in the JSON file: title, type and properties. Properties is similar with attributes. The properties can be one or more than one. Based on Figure 5, a<sub>1</sub> until a<sub>n</sub> is attribute for each object.

Definition 2: Let, JSON file,  $A = \{S,O,X\}$ , where S is represent a title, O is represent object name, and X is represent properties or attributes.

Definition 3: Each O can have more than one object, which is  $a_1$  until  $a_n$  is number of objects in JSON file.

$$F(G) = O \{a_1, a_2, a_3, a_n\}$$

Definition 4: Each X can have one or more properties, which is  $a_i$  until  $n_i$  is properties of X.

$$F(H) = X \{b_1, b_2, b_3, b_n\}$$

According to Figure 6, three elements involve in JSON: title, type and properties. Title is represented object name; type is represented object type; and properties is represented details about the object.

```
{
  "Title": "Schema"
  "type": "O"
  "properties": {
    "b1": { "type": "string" },
    "b2": { "type": "string" },
    .
    .
    "bn": { "type": "string" }
  },
}
```

Figure 6. JSON schema

## 5. RESULTS

In this section, the performance of AJAX and JQuery using matching algorithm will be analysed in term of time execution and memory usage. We evaluate the performance of the accessing system at the concurrent time. The systems are build using a personal computer equipped with 2.40 GHz Intel(R) Core (TM) i7-5500 CPU, 8.00 GB and a 250 GB solid-state drive. The operating system is Microsoft Windows 10. We are using existing number of patients (1294) according to Table 1 to test the performance of AJAX and JQuery based on matching algorithm in Figure 7.

### 5.1. Modeling process

Four components involve in modelling process. There is graphical user interface (GUI), AJAX/JQuery, hybrid algorithm, and data model.

- Graphical user interface (GUI). A simple GUI as shown in Figure 7 are designed to allow users choose any related symptoms before making a submission. First process is submission. Once submission is done, the next process AJAX/JQuery.
- AJAX/JQuery. Figures 8 and 9 shows a simple function using AJAX and JQuery. In AJAX, value of x is variable, value of y is representing a method; post and value of z is representing as a URL/destination; *send\_data.php*. Meanwhile, value of x represents target URL; *send\_data.php* and value of y represent method; post for JQuery.
- Hybrid algorithm. Figure 10 shows a program structure based on hybrid algorithm in Figure 2. Based on this program structure, first steps are initial declaration of symptoms ( $s_1, s_2, s_n$ ) and list of data from users ( $data_1, data_2, data_n$ ). Looping process using *for* will be used to calculate possible percentage based matching symptoms and weightiest. Finally, this program structure will display a result (in percentage) as a *total*.
- Data model. This section explained the implementation of XML and JSON as a data model. Figures 11 and 12 shows a data model in XML and JSON.

Figure 7. GUI form based on COVID-19 symptoms

```

Function targetURL () {
  // initial declaration variable
  let x = new XMLHttpRequest();
  // target URL
  http.open(post,send_data.php,true);
  // execute URL
  http.send();
}

```

Figure 8. AJAX approach

```

$(function() {
  $("#form").submit(function(e)
  e.preventDefault();
  $.ajax({
    url: 'send_data.php',
    type: 'post',
    data: $("#form").serialize(),
    success: function(data) {
      // message success/fail
    }
  });
});

```

Figure 9. JQuery approach

```

$num = $n // let say $n equal number of symptoms
$s = array['symptoms']; // $s1, $s2, $sn
$data = $_POST['input']; // data1, data2, data_n
$total = 0;
for ($i=0; $i<count($_POST['input']); $i++) {
  for ($j=0; $j<$num; $j++) {
    if ($data[$i]!="") {
      $total = $total + ($data[$i] * $s[$j]);
    }
  }
}
$total; // display percentage as a result

```

Figure 10. Program code based on MA

```
<List>
<Age>...</Age>
<Gender>...</Gender>
<Contact>...</Contact>
<Gender>...</Gender>
<Fever>...</Fever>
<Cough>...</Cough>
<Myalgia>...</Myalgia>
</List>
```

Figure 11. XML data model

```
[{"Age":"","Gender":"","Contact":"","Gender":"L","Fever":"","Cough":""}, {"Myalgia":"","Age":"","Gender":"","Contact":"","Fever":"","Headache":""}, ...]
```

Figure 12. JSON data model

**5.2. Comparative analysis**

Table 3 shows the comparative performance among different approaches based on 1294 users' access at the concurrent time. Two types of measurement performance will be evaluated: time execution and memory usage. First, according to Table 3, time execution of approach V based on XML is reduce around 10-15% compared to approach I and III. Meanwhile, time execution of approach VI based on JSON is reduce around 5-15% compared to approach II and IV. In this case, approach VI is better compared to other approaches in term of time execution.

In term of memory usage, approach V and VI using less memory usage compared to other approaches. In this case, performance of approach V and VI is better compared to other approaches because of execution concurrent users (1294) at the same time using less memory but still can produce the output ether positive or negative COVID-19 based on hybrid algorithm. Figures 13 and 14 represent time execution and memory usage comparison among six (6) approaches. Based on Figures 13 and 14, approach VI (JQuery + JSON + hybrid algorithm) is clearly between compared to other approaches which is fast time execution time and use less memory.

Table 3. Time execution and memory usage

Approach	Parameters	Time execution (ms.)	Memory usage (%)
I	AJAX + XML	0.03335	0.00960
II	AJAX + JSON	0.01195	0.00925
III	JQuery + XML	0.02635	0.00860
IV	JQuery + JSON	0.01105	0.00965
V	JQuery + XML + Hybrid Algorithm	0.02305	0.00785
VI	JQuery + JSON + Hybrid Algorithm	0.00900	0.00695

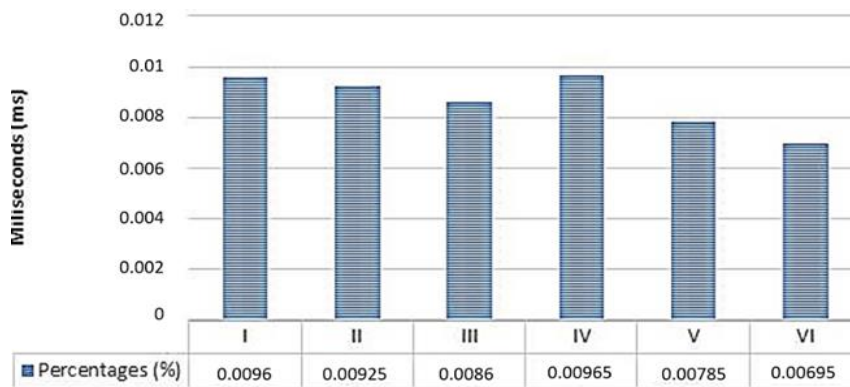


Figure 13. Time execution performance



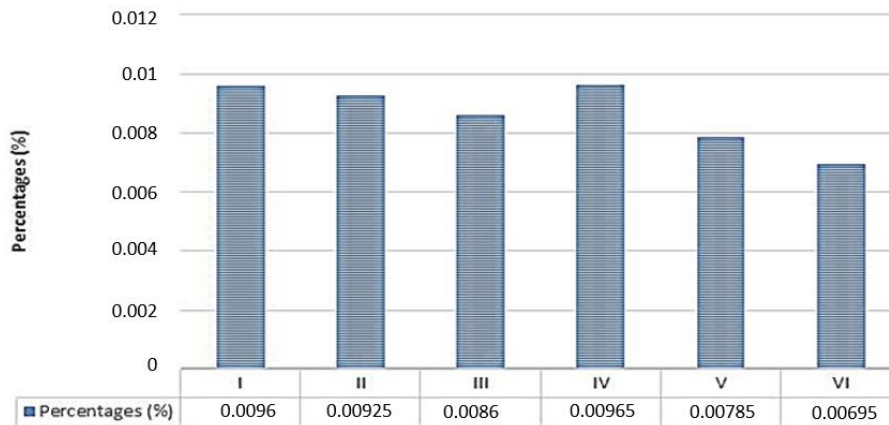


Figure 14. Memory usage performance

## 6. CONCLUSION

In conclusion, this study had successfully proposed and implemented an efficient storing and retrieving data approach to execute a large amount of data at concurrent time access. The approach using JQuery + JSON + hybrid algorithm had fast time execution time and uses less memory is significant to analyse the data of COVID-19 online screening based on its symptoms using a keyword search algorithm. The results proved that the JQuery + JSON + hybrid algorithm was the efficient storing and retrieving data approach to execute a large amount of data at concurrent time access compared to the AJAX + XML, AJAX + JSON, JQuery + XML, JQuery + JSON, and JQuery + XML + hybrid algorithm. The findings of this study had supported those of the previous ones proving the JQuery + JSON + hybrid algorithm as the best approach in storing and retrieving data compared to the others method. The processes of initialising the user input and weightiest symptoms were several of the processes involved in the hybrid algorithm. A Hybrid algorithm process also explained the execution of mapping, value and total data model. The limitation of this study is a narrow focus. This study focused only on a few various cohorts of the patient. Further study in the future may involve many different cohorts and a large number of samples to provide more precise study findings.




## REFERENCES

- [1] P. Bablani, Y. Shamsi, P. M. Kaapor, and M. Sharma, "COVID-19 & Effectiveness towards its available treatment," *International Journal of Research and Analytical Reviews*, vol. 7, no. 2, pp. 429-434, April. 2020, doi: 10.1729/Journal.23479.
- [2] H. Harapan, et al., "Coronavirus disease 2019 (COVID-19): A Literature Review," *Journal of Infection and Public Health*, vol. 13, no. 5, pp. 667-673, May. 2019, doi: 10.1016/j.jiph.2020.03.019.
- [3] R. Lu, et al., "Genomic characterisation and epidemiology of 2019 novel coronavirus: implications for virus origins and receptors binding," *Lancet* 2020, vol. 395, pp. 565-574, Feb 2020, doi: 10.1016/S0140-6736(20)30251-8.
- [4] C. J. Burrell, C. R. Howard, and F. A. Murphy, "White's medical virology. 5<sup>th</sup> ed," United States: *Academic Press*, pp. 557-583, Nov. 2016.
- [5] WHO. "Novel coronavirus situation dashboard. Geneva," *Switzerland: WHO*, 2020.
- [6] <https://www.who.int/news-room/q-a-detail/q-a-coronaviruses>. March 2020. Last accessed 24 March 2020.
- [7] F. He, Y. Deng, and W. Li, "Coronavirus disease 2019 (COVID-19): What we know?," *Journal of Medical Virology published by Wiley Periodicals Inc*, vol. 92, no. 7, pp. 719-725, 2020, Jul. 2020, doi: 10.1002/jmv.25766.
- [8] J. Yang et al., "Prevalence of comorbidities in the novel Wuhan coronavirus (COVID-19) infection: a systematic review and meta-analysis," *International journal of infectious diseases*, vol. 94, pp. 91-95, May. 2020, doi: 10.1016/j.ijid.2020.03.017.
- [9] B. Li et al., "Prevalence and impact of cardiovascular metabolic diseases on COVID-19 in China," *Clinical Research in Cardiology*, vol. 109, pp. 531-38, Mac. 2020, doi: 10.1007/s00392-020-01626-9.
- [10] L. Q. Li et al., "COVID-19 patients' clinical characteristics, discharge rate, and fatality rate of meta-analysis," *Journal of Medical Virology*, vol. 92, pp. 577-583, Mac. 2020, doi: 10.1002/jmv.25757.
- [11] T. Singhal, "A Review of Coronavirus Disease-2019 (COVID-19)," *The Indian Journal of Pediatrics*, vol. 87, no. 4, pp. 281-286, Mac 2020, doi: 10.1007/s12098-020-03263-6.
- [12] P. Sun, X. Lu, C. Xu, W. Sun, and B. Pan, "Understanding of COVID-19 based on current evidence," *Journal of Medical Virology*, vol. 92, no. 6, pp. 548-551, Jun. 2020, doi: 10.1002/jmv.25722.
- [13] W.-J. Guan et al., "Clinical Characteristics of Coronavirus Disease 2019 in China," *The New England Journal of Medicine*, vol. 58, no. 4, pp. 711-712, Apr. 2020, doi: 10.1016/j.jemermed.2020.04.004.
- [14] N. Chen et al., "Epidemiological and clinical characteristics of 99 cases of 2019 novel coronavirus pneumonia in Wuhan, China: a descriptive study," *The Lancet*, vol. 395, no. 10223, pp. 507-513, Jan. 2020, doi: 10.1016/S0140-6736(20)30211-7.
- [15] H. Shi et al., "Radiological findings from 81 patients with COVID-19 pneumonia in Wuhan, China: a descriptive study," *Lancet Infect*, vol. 20, no. 4, pp. 425-434, Apr. 2020, doi: 10.1016/S1473-3099(20)30086-4.
- [16] C. Huang et al., "Clinical features of patients infected with 2019 novel coronavirus in Wuhan, China," *The Lancet*, vol. 395, no. 10223, pp. 497-506, Feb. 2020, doi: 10.1016/S0140-6736(20)30183-5.




- [17] Yang X *et al.*, "Clinical course and outcomes of critically ill patients with SARS-CoV-2 pneumonia in Wuhan, China: a single-centered, retrospective, observational study," *Lancet Resp Med* 2020, vol. 8, no. 5, pp. 475-481, May. 2020, doi: 10.1016/S2213-2600(20)30079-5.
- [18] E. Rafiq, M. W. El-Kharashi, and F. Gebali, "A fast string search algorithm for deep packet classification," *Computer Communication*, vol. 27, no. 15, pp. 1524-1538, Sept. 2004, doi: 10.1016/j.comcom.2004.06.005.
- [19] R. M. Karp and M.O Rabin, "Efficient Randomized Pattern-Matching Algorithms," *Ibm Journal of Research and Development*, vol. 31, no. 2, pp. 249-260, Mar. 1987, doi: 10.1147/rd.312.0249.
- [20] W. Yang, "Mealy machines are better model of lexical analyzers," *Computer Language Journal*, vol. 22, no. 1, pp: 27-38, Apr. 1996, doi: 10.1016/0096-0551(96)00003-3.
- [21] F. J. Franek, C. G. Jennings, and W. F. Smyth, "A simple fast hybrid pattern matching algorithm," *Journal of Discrete Algorithms*, vol. 5, no. 4, pp: 682-695, Dec. 2007, doi: 10.1016/j.jda.2006.11.004.
- [22] S. Hakak, A. Kamsin, P. Shivakumara, A. Gilkar, W. Z. Khan, and M. Imran, "Extact String Matching Algorithms: Survey, Issues, and Future Research Directions," *IEEE Access*, vol. 4, no. 72, pp. 144-156, Aug. 2019, doi: 10.1109/ACCESS.2019.2914071.
- [23] A. M. Al-Ssulami, "Hybrid string matching algorithm with a pivot," *Journal of Information Science*, vol. 41, no. 1, pp. 82-88, Feb. 2015, doi: 10.1177/0165551514555668.
- [24] A. Marchetto, P. Tonella, and F. Ricca, "State-Based Testing of AJAX Web Applications," *1st International Conference on Software Testing, Verification, and Validation*, vol. 1, Jun. 2008, pp. 1-10, doi: 10.1109/ICST.2008.22.
- [25] C. Gyorodi *et al.*, "Web 2.0 Technologies with jQuery and AJAX. 2009," *Computer Technology and Computer Programming*, vol. 1, Sept. 2012, doi: 10.1201/b13124-7.
- [26] J. J. Li and C. L. Peng, "jQuery-based AJAX general interactive architecture," *2012 IEEE International Conference on Computer Science and Automation Engineering*, Aug. 2012, doi: 10.1109/ICSESS.2012.6269466.
- [27] A. Balmin and Y. Papakonstantinou, "Storing and querying XML data using denormalised relational databases," *VLDB J*, vol. 14, no. 1, pp. 30-49, Mac. 2005, doi: 10.1007/s00778-003-0113-1.
- [28] F. Zhang and Z. Ma, "Representing and reasoning about XML with ontologies," *Applied Intelligence*, vol. 40, no. 1, pp. 74-106., Jan. 2014, doi: 10.1007/s10489-013-0446-4.
- [29] M, K. Yusof and M. Man, "Efficiency of JSON approach for data extraction and query retrieval," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 4, no. 1, pp. 203-214, Oct. 2016, doi: 10.11591/ijeecs.v4.i1.pp203-214.
- [30] M. Enoki, J. Sameon, H. Horii, and M. Hirzel, "Event processing over a distributed JSON store," *Web Information Systems Engineering – WISE*, vol. 8787, pp. 395-404, 2014, doi: 10.1007/978-3-319-11746-1\_28.

## BIOGRAPHIES OF AUTHORS






**Mohd Kamir Yusof**    received his PhD in Computer Science at Universiti Malaysia Terengganu (UMT) Terengganu in May 2021. He received her master's degree in Master of Science (Computer Science) from Universiti Teknologi Malaysia (UTM) Skudai, Johor in 2008 prior to finishing her study in Bachelor's degree also in the same stream from Universiti Teknologi Malaysia (UTM) Skudai, Johor in 2006. His research is focused on development of mobile application, web-based application, and data integration. He can be contacted at email: mohdkamir@unisza.edu.my.



**Wan Mohd Amir Fazamin Wan Hamzah**    currently is a lecturer of Computer Science at Universiti Sultan Zainal Abidin, Malaysia. He received his Phd and MSc in Computer Science from Universiti Malaysia Terengganu in 2016 and 2010. His main research areas are in the fields of learning analytics, e-learning, gamification, and cloud computing. He can be contacted at email: amirfazamin@unisza.edu.my.



**Nur Shuhada Md Rusli**    received the B.Sc. degree in computer science from the University Sultan Zainal Abidin, Terengganu, Malaysia. She is currently a Lecturer with the Department of Information and Communication, Politeknik Balik Pulau, Pualau Pinang, Malaysia. She has supervised and co-supervised more than 20-degree students. Her research interests include system development, internet of things, and intelligent systems. She can be contacted at email: nurshuhada@pbu.edu.my.