JOIN (Jurnal Online Informatika)
p-ISSN: 2528-1682, e-ISSN: 2527-9165
Volume 11 Number 1 | June 2026: 197-205
DOI: 10.15575/join.v11i1.1653

Using Readability Metrics in Estimating the Readability of REpresentational State Transfer State Transfer Uniform Resource Identifiers Schema

Fuad Alshraiedeh [1], Norliza Katuk [2], Hossam Almahasneh [3]
[1]Department of Software Engineering, Zarqa University, Jordan
[2] School of Computing, Universiti Utara Malaysia, Malaysia
[3]Department of Computer Science, Zarqa University, Jordan

Article Info

Article history:

Received May 19, 2025
Revised June 6, 2025
Accepted July 13, 2025
Published April 30, 2026

Keywords:

Readability methods
URI readability
URI schema

ABSTRACT

Uniform Resource Identifiers (URIs) may have a direct impact on the understanding of REpresentational State Transfer State Transfer (RESTful) functionality, and thus, on the discovery of final RESTful product. RESTful Web Services (WS)/Application Programming Interfaces (APIs) are designed to expose data and functionality through resources accessed by dedicated URIs over HyperText Transfer Protocol (HTTP), which recently represents the direct descriptions schema of what functions does the concerned RESTful WS/API present. Furthermore, the discovery of suitable RESTful is heavily rely on the simplicity of understanding their URI schemas, which recently suffer from critical issues in how to measure their readability. For that, WS/APIs developers aspire to measure the readability of RESTful URI schemas before exposing them over the Internet to estimate their usability. Consequently, this research proposes four readability metrics for the stated purpose namely: Flesch-Kincaid (F-K), Flesch Reading Ease (FRES), Simple Measure of Gobbledygook (SMOG), and Coleman Liau Index (CLI). The research identifies the variables required to calculate the readability metric and formulate the equations for them. Four experts in linguistics were asked to validate the proposed metrics and their identified variables. The research successfully conducted empirical research on 8 well-known RESTful WSs/APIs of the dataset, and the proposed metrics were implemented on 6952 URIs schemas. The average values for the aforementioned metrics were 7.41%, 59.63%, 6.73%, and 17.55% respectively, where in certain metrics, a low average value signifies easy readability, but in others, it signifies hard readability, and vice versa.

Corresponding Author:

Fuad Alshraiedeh
Software Engineering, Faculty of Information Technology, Zarqa University, Jordan
Email: falshraiedeh@zu.edu.jo

1. INTRODUCTION

Cloud computing platforms and information technology (IT) companies, like *LinkedIn*, *BestBuy*, *Twitter*, *Facebook*, *OpenStack*, and *Google Cloud Platform*, have paid close attention to information systems and software applications that use APIs and WSs [1]. In order to facilitate the greatest possible number of external users accessing their resources over the Internet, these companies provide their services using the RESTful architecture style [2]. Nowadays, RESTful-style architecture is considered popular due to its low cost, ease of use, and flexibility [3].

The RESTful-style is built on well-known Internet protocols and has lower network needs compared to the Simple Object Access Protocol (SOAP) architecture and Remote Procedure Call (RPC) [4]. These properties make the RESTful-style easier and more interoperable than any other contemporary web architecture style [5]. In recent years, the RESTful-style has been created and implemented in a wide range of information systems [6]. As a result, the majority of developing modular distributed software systems have been moved to the RESTful architecture [7]. In a distributed software system, system resources are exposed as Internet services described by URIs and accessed using the Hypertext Transfer Protocol (HTTP) [8]. Furthermore, well-named URI nodes may be easier to read and understand by their consumers (i.e., developers and end-users), which would ease their adoption and facilitate the task of their discovery [9]. However, URI readability and understandability are the two most important factors in corresponding RESTful quality and discovery [10]. As a result, developers and designers must consider both good and harmful practices when naming and documenting URI nodes during the coding and design phases.

Software developers utilize related natural language words to name software entities (such as identifiers, resources, variables, etc.). These entities have been found to have a major impact on the final software product's quality and discoverability. Linguistic relationships between API and WS entities are definitive in RESTful style; consequently, the absence of such poor naming as well as a linguistic relationship may hinder RESTful discovery. Palma et al. (2015) and Aghajani et al. (2018) described best and bad practices for naming software entities as linguistic patterns and anti-patterns[11], respectively. Furthermore, amorphous and ambiguous resource names are linguistic anti-patterns that describe URI nodes using untidy and unrepresentative words. The readability of URIs provides an overview of corresponding API/WS functionality, which is an essential step to ensure that it gets a better rate of discovery and use. However, based on our knowledge, there is no defined methodology for measuring the readability of RESTful URI schemas, and there is no reliable metrics for this purpose.

In this research, we aim to measure the readability of RESTful URI schemas by calculating four proposed readability metrics, and over time, can help developers improve their naming skills. MS-Excel functions were used to compute the required variable values. Before starting the experiments, the researcher identifies the variables required to calculate the readability metric and formulates the equations that calculate them. Thus, an MS-Excel sheet was generated for each RESTful in the dataset. Due to a large number of URI schemas in the dataset, it is impossible to report all schemas' readability values; therefore, the research reported the average values of each metric for corresponding RESTful URI schemas.

Moreover, the research presents a compilation of readability methods and tools that we have reviewed during our research. Different academics in linguistics have developed methods to measure the readability of various textual materials. Our analysis revealed that the majority of the formulas assessed readability by considering the grading level according to the US grading system. The grading level at both primary and secondary levels is subject to variation based on geographic location and environment. However, there remains an unaddressed question regarding the global validity of these methods. For example, Djoko formula, Fernandez Huerta Index, Kandal and Moles Index, and Al-Heeti grade level are several methods created by researchers to measure the readability of Indonesian, Spanish, French, and Arabic texts [12][13][14], respectively. However, the mentioned methods were applied to texts, paragraphs, sentences, etc., of natural languages but not to the URI schema, which is the core of this research. The formula devised for non-English text demonstrated a significant correlation with individuals' existing reading abilities. However, the scores obtained from methods designed for the English language did not align with the anticipated results. Table 1 lists a sample of readability methods that have been presented in the literature.

Table 1. Sample of readability methods

No	Readability method	Description
1.	Flesch Reading Ease Readability	One of the accurate measures for school text. Higher the score, text is easily readable, difficulty increases with decrease in score level. Score ranges from 0 to 100.

No	Readability method	Description
2.	Dale-Chall Read-ability	Inspired by FRE, considers difficult words and sentence length to result Dale-chall score. A score below 4.9 for a text is treated as easily understood by Grade 4 student in US and grade above 10 for college graduates.
3.	Flesch- Kincaid Grade Level	Modified version of FRE, Score indicates the grade-school level education required to understand textual content in US.
4.	FOG	Research based on daily newspapers, magazines is the origin of this formula, text with score 7–8 Fog Index is considered as ideal and score above 12 is too hard for most of the people.
5.	Automatic Readability Index	Outputs an approximate grade level required to comprehend the text, for example US grade Level 1 corresponds to understandable to age 6 to 8 and grade 12 corresponds to 17 years old. It is based count of characters and words.
6.	CLI	It approximates US grade level to understand the text based on characters instead of syllables. A grade level of 10.6 is easily understood by 10–11 grade student whereas 14 is for college level student.
7.	SMOG	SMOGpredicts two grades higher than Dale-Chall formula. It is considered appropriate for secondary age. Atext with polysyllabic count 1–6 falls under grade level 5 whereas 211–240 results a grade of 18.
8.	PSK	Best formula to calculate text sample for US grade level, suited best for primary age children and advised not to use for text for children above age of 10.
9.	Read-X	Read-X does real-time readability study of web content through using web search and filtering depending on category level. It then categorizes the search results according to their themes.
10.	Spache	Spache is similar to Dale-Chall readability formula but not ideal for advanced texts

The majority of the previously published readability formulas took into account criteria/attributes such as word length, sentence length, total syllables, and the number of complex words. However, it is important to note that these criteria/attribute have certain restrictions that may lead to certain outcomes. The developed methods, such as Lexile Framework, ATOS, Read-X, Coh-Metrix, and the new Dale-Chall readability formula, occurred after the 1980s. These methods assess readability by considering criteria such as cognitive-structural elements, semantic units, and syntactic structure complexity. However, according to Pawan Kumar Ojha (2021), he categorized the common attributes that have a direct impact on the readability level as summarized in Table 2.

Table 2. The Attributes of Readability Metrics

No	Readability method	Description
1.	Sentence	FRE, Dale-Chall readability, FKGL, Gunning Fog Index, ARI, FRY, PSK, SPACHE, CLI, Raygor Estimate Graph, Djoko Formula, Pisarek's Index, Fernandez Huerta Index, Kandal and Moles Index, Al-Heeti Grade Level, The Lexile Framework
2.	Syllables	FRE, FKGL, FRY.
3.	Words	FRE, Dale-Chall readability, FKGL, Gunning Fog Index, ARI, PSK, CLI.
4.	Unfamiliar Words	Dale-Chall readability
5.	Number of polysyllables word count	SMOG
6.	Character count	ARI, CLI, BRI, Gunning Fog Index

The structure of a URI scheme differs from the structure of natural language in terms of linking words together to form sentences and paragraphs, where the URI schema identifiers are linked together only through the symbol \ which is considered one of the natural languages' symbols, and has meanings depending on its usage context. So, there is a need to bridge the gap between the structure of URI schemas and the structure of natural language, and find a to find roadmap to apply natural language readability metrics on RESTful URI schema, in order to help RESTful developers estimate the usability of WSs/APIs, which is the contribution of this research.

In the field of this research, it is important to take into account other criteria; furthermore, the nature of readability depends on only the nodes that have been structured in the RESTful URI schemas, which means that the number of sentences, syllables, and words are too limited, so that we rely on experts in linguistics to overcome this challenge.

2. METHODOLOGY

The initial step of this section is to define the three readability metrics to identify the input of each metric. These metrics were chosen because they are simple to calculate, are well-known in the readability field, and are provided by Mashape.com. Further, these metrics were implemented on XML schemas (i.e., WSDL) [15], and this prompted the researcher to apply them to URI schemas. Experts in linguistics confirmed that these metrics are appropriate for the desired goal, and they were also asked to identify from the URI schemas what variables are required to calculate each metric.

2.1. Flesch-Kincaid (F-K)

The F-K readability tests were designed to assess how difficult it was to understand an English reading text, where the text is any stretch of language that can be understood in the context [16], [17], [18]. It may be as simple as one or two words (such as a stop sign) or as complex as a novel [19]. The F-K readability level presents a score as a reader's ability level, making it easier for designers, high-level students, end-users, and others to judge the readability level of various texts [20]. The readability level was calculated by Equation 1 [15].

$$0.39 * \frac{(Total\ of\ Words)}{(Total\ of\ Sentences)} + 11.8 * \frac{(Total\ of\ Syllables)}{(Total\ of\ Words)} - 15.5 \quad (1)$$

Equation 1 depends on the average number of words per sentence and the average number of syllables per word. In this research, the number of sentences represents the number of URIs, whereas the number of syllables represents the number of nodes per URI. Each URI for the entire dataset was analysed independently to extract the required variable values for the F-K calculation. Moreover, the higher the value of F-K, the higher the grade level for reading and understanding [21].

The higher the URI's F-K value, the more difficult it is to read and understand, while the lower the value is the lower the required level to read and understand RESTful URI schemas. In other words, a lower-level F-K value is a higher opportunity of corresponding to a RESTful to discovery [22] [20].

2.2. Flesch Reading Ease (FRES)

In the FRES test, higher values indicate text material that is easier to read and understand, whereas lower values mark passages that are more difficult and complex to read and understand [19][23]. The FRES score test was calculated using Equation 2 [15]:

$$206.835 - 1.015 * \left(\frac{Total\ of\ Words}{Total\ of\ Sentences}\right) - 84.6 * \left(\frac{Total\ of\ Syllables}{Total\ of\ Words}\right) \quad (2)$$

The lower values of FRES indicate a text that is more difficult to read and more complex to understand. Equation 2 showed the required variables to calculate FRES. A short node or sentence containing a few syllable words makes up the URI with a very high FRES. Conversely, the URI with a very low FRES value consists primarily of very long nodes and many ambiguous words. However, the lower the URI FRES value, the lower the discovery opportunity. Moreover, it provides a value between 0 and 100. A value of 100 means the URI schema is straightforward to read and understand. A value of 0 means the URI schema is very difficult to read and complex to read [24][18].

2.3 Simple Measure of Gobbledygook (SMOG)

The SMOG grade measured readability and estimated the years of education needed to understand a piece of text [20]. Three steps were needed to calculate SMOG: counting the number of sentences (1 sentence per URI), counting the polysyllables in these sentences (words of three or more syllables), and finally, calculating the SMOG value using Equation 3 [15]:

$$1.043 * \sqrt{Total\ of\ Polysyllabes * \frac{30}{Total\ of\ Sentences}} * 3. \quad (3)$$

This metric measures readability and estimates the amount of education needed to understand a piece of text [15]. In this research, the RESTful users belong to different levels of education and could be RESTful designers or general users, so the URI schema must be readable and understandable for the widest range of education levels as much as possible. The SMOG depends on the number of sentences and the number of polysyllables, where polysyllables are counted with words of 3 or more syllables [24]. Moreover, the value of SMOG indicates the required education level to understand a piece of text, where a higher value means it is more difficult to understand and requires a higher level of education, while a low value means it is easy to understand, which requires a lower level of education [25][17]

2.4 Coleman Liau Index (CLI)

The CLI is designed to be easily calculated from a sample of texts. In addition, unlike syllable-based readability indexes, it does not require analyses of the character content of words; only their length in characters is required [15]. The CLI was calculated using Equation 4 [15][26]:

$$0.0588 * L - 0.296 * S - 15.8 \quad (4)$$

Where L is the average number of letters per 100 words, and S is the average number of URI schema nodes per. However, CLI is calculated once for all RESTful URI schemas, which means each RESTful URI has only one CLI value regardless of the number of its URIs.

The CLI was designed to be easily calculated mechanically from samples of hard-copy texts [27]. Unlike syllable-based readability indexes, it does not require analyses of the character content of words; rather, only their length in characters is required [15]. It gauges the understandability of a text, and its output approximates the education level thought to be necessary to understand the text [23][19]. Moreover, CLI is directly calculated from a sentence, paragraph, or text document. Unlike syllable-based readability indices, it does not require that the character content of words be analysed; rather, only their length in characters is required. The CLI is calculated by Equation 4, where the average number of letters per 100 words (L) and the average number of sentences per 100 words (S) are required. However, CLI is calculated once for each RESTful, which means L and S average values are calculated for the whole URI schemas in the RESTful.

3. RESULT AND DISCUSSION

Before starting the experiments, the researcher identifies the variables required to calculate the readability metric and formulates the equations that calculate them. Thus, MS-Excel functions were used to compute the required variable values, and an MS-Excel sheet was generated for each RESTful in the dataset. The purpose of these variable values is to calculate the readability metrics for each URI schema in the RESTful. Table 3 shows the required variable values and the formulas of MS-Excel functions that were used to compute them.

Table 3. The Formulas of MS-Excel Functions for the Required Variable Values

The Required Variable	Description
Total of Words for each node	=IF(LEN(TRIM(x1))=0,0,LEN(TRIM(x1))-LEN(SUBSTITUTE(x1," ",""))+1). Where x1 represents the node's cell name
Total of Words for each URI Schema	=SUM(x1:xn). Where x1 : xn represent the range of the cells for such URI Schema's nodes
Total of Sentences	=SUM(COUNTA(x1),.....,COUNTA(xn)). Where x1 and xn represent the range of the cells for such URI Schema's nodes
Total of Polysyllables	=IF(xn>3,1,0). Where xn represents the cell's name that contain the total number of words for each URI schema
Total of Syllables	=COUNTA(x1:xn). Where x1 : xn represent the range of the cells for such URI Schema's nodes

p-ISSN: 2528-1682
e-ISSN: 2527-9165

Due to the large pool of RESTful, there is no obvious reason to choose one over another, and each RESTful has an equal chance of being selected as a subject. Therefore, this research employed a simple random sampling method to collect the dataset. It collected the primary dataset from various repositories for a representative dataset of RESTful. It relied on our previous study, where the dataset is available on GitHub at https://github.com/Fuad81/rest-api. Furthermore, the dataset was collected from 5 publicly available RESTful WSs and APIs, with 6952 URI schemas. Table 4 shows online documentation of the selected RESTful that made up the dataset.

Table 4. List of 8 RESTful APIs and WSs with their URL

The Required Variable	The Function Formula
BestBuy	www.bestbuy.com/documentation
Bitly	www.dev.bitly.com/api.html
CharlieHarvey	www.charlieharvey.org.uk/about/api
Dropbox	www.dropbox.com/developers/core/docs
Externalip	www.api.externalip.net

This research used a dataset with 8 publicly available RESTful APIs, consisting of 6952 URI schemas with 21342 corpus nodes. The nodes were analysed, and their syllables were counted, yielding a total of 25832 words, and the readability metrics were calculated for each URI schema in RESTful. Due to a large number of URI schemas in the dataset, it is not possible to report all schemas' readability metric values; therefore, the research used the average of the readability values for the corresponding RESTful URI schemas.

The study used MS-Excel functions such as SUM, SQRT, AVERAGE, and ABS to calculate the readability values for all URI schemas. Furthermore, the study used the MS-Excel formula to implement the readability metrics equations, namely Equation 1, Equation 2, and Equation 3.

The study calculated the readability metrics for each URI schema in the dataset, and only the average value for each metric is shown in this research. Moreover, the four metrices were chosen based on the recommendation from experts in linguistics, where the metrices' required variables can be edged, and then extracted from the RESTful URI schema. The following example shows how to calculate the readability metrics F-K, FRES, SMOG, and CLI, respectively, for each URI schema in the RESTful:

Example - The MS-Excel function formulas presented in Table 6 were implemented on one of the Dropbox RESTful URI schemas. The results are shown in Table 5:

Table 5. The Required variable values for the readability metrics

Total of Words	Total of Sentences	Polysyllable	Total of Syllables
6	4	1	4

Based on these variables' values, the readability metrics can be calculated as the following:

The F-K metric is calculated using Equation 1.

$$0.39 * \frac{(Total\ of\ Words)}{(Total\ of\ Sentences)} + 11.8 * \frac{(Total\ of\ Syllables)}{(Total\ of\ Words)} - 15.5$$

F-K $= 0.39 * \frac{6}{4} + 11.8 * \frac{4}{6} - 15.5$
= 0.59 + 7.87 – 15.5
= -7.13 The absolute value
F-K = 7.13

The FRES metric is calculated using Equation 2.

FRES $= 206.835 - 1.015 * \left(\frac{Total\ of\ Words}{Total\ of\ Sentences}\right) - 84.6 * \left(\frac{Total\ of\ Syllables}{Total\ of\ Words}\right)$
$= 206.835 - 1.015 * \frac{6}{4} - 84.6 * \frac{6}{4}$
= 206.835 − 1.52 – 126.9

FRES = 78.92

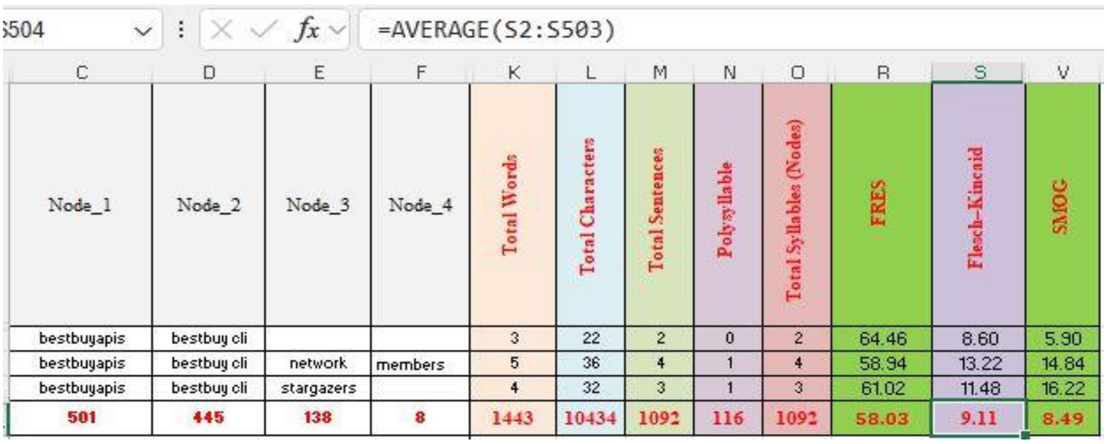
The SMOG metric is calculated using Equation 3.

$$SMOG = 1.043 * \sqrt{Total\ of\ Polysyllabes * \frac{30}{Total\ of\ Sentences}} + 3.1291$$
$$= 1.043 * \sqrt{4 * \frac{30}{4}} + 3.1291$$
$$= 5.71\ + 3.1291$$
$$SMOG\ = 8.84$$

The CLI metric is calculated using Equation 5.

$$\text{CLI} = 0.0588 * L - 0.296 * S - 15.8$$
$$= 0.0588 * L - 0.296 * S - 15.8$$
$$= 0.0588 * 10.78 - 0.296 * 25.94 - 15.8$$
$$= 0.63 - 7.68 - 15.8$$
$$= -22.85 \text{ The absolute value}$$
$$\text{CLI} = 22.85$$

The example above shows the calculation of readability metrics for one URI schema. However, the calculation was performed on the first URI schema in each RESTful and the calculations for the remaining URI schemas were made using MS-Excel features, namely DRAG TO EXTEND SERIES. Then, the research computes the average value for each metric for each RESTful URI schema in the dataset. The average values were computed by the MS-Excel AVERAGE function, as shown in the screenshot of the Bestbuy sheet in Figure 1. The average readability metrics for the RESTful in the original dataset are shown in Table 6.


S504 =AVERAGE(S2:S503)

Node_1	Node_2	Node_3	Node_4	Total Words	Total Characters	Total Sentences	Polysyllable	Total Syllables (Nodes)	FRES	Flesch–Kincaid	SMOG
bestbuyapis	bestbuy cli			3	22	2	0	2	64.46	8.60	5.90
bestbuyapis	bestbuy cli	network	members	5	36	4	1	4	58.94	13.22	14.84
bestbuyapis	bestbuy cli	stargazers		4	32	3	1	3	61.02	11.48	16.22
501	445	138	8	1443	10434	1092	116	1092	58.03	9.11	8.49

Figure 1. The average values of the readability metrics for the Bestbuy URI schemas

Table 6. The average values of the readability metrics for the original Dataset

RESTful	F-K	FRES	SMOG	CLI
BestBuy	9.11	58.03	8.49	19.10
Bitly	7.54	61.14	2.3	22.64
CharlieHarvey	7.39	67.61	7.9	22.10
Dropbox	5.81	72.99	8.15	22.51
Externalip	8.74	69.52	8.85	17.00
Facebook	6.73	32.41	5.63	15.84
GoogleBook	6.71	72.84	6.87	12.25
Average	**7.41**	**59.63**	**6.73**	**18.80**

As shown in Table 6, the average values for F-K, FRES, SMOG, and CLI were 7.41%, 59.63%, 6.73%, and 18.80%, respectively. In more detail, the F-K level for BestBuy URI schemas represents the most difficult URI schemas to read and understand, where its value was the highest among the rest of the RESTful. On the other hand, the level of Dropbox was the easiest to read and understand in the original URI schemas. In FRES, the level comparison is the opposite, whereby the lower level is harder to read and understand. Most of the readability levels in the URI schemas were similar, except for Facebook. In the SMOG metric, BestBuy and ExternalIP URI schemas were the harder levels of reading and understanding, while Bitly schemas were the lower levels. schema, while Instagram was the lowest level. GoogleBook was the lowest level of CLI in the original URI schemas, so it was the easiest to read and understand, whereas Bitly was the hardest.

4. CONCLUSION

The goal of this research is to propose metrics to measure the readability of RESTful URI schema. Four readability metrics were used for this purpose, namely, F-K, FRES, SMOG, and CLI. However, the proposed metrics were implemented on 8 well-known RESTful WSs/APIs of the dataset with 6952 URI schemas. Although the results of the research suggested an acceptable level of accuracy for pattern prediction, with 82.61% of recall, 97.34% of precision, and 88.41% of F-measure. This research shows how this technique can contribute to facilitating the readability and understandability of RESTful URI nodes. In the future, this research aims to extend the proposed prediction technique to address more types of anti- patterns, and also predict the patterns for different types of RESTful documentation, such as Web application description and Swagger. Moreover, another research direction is to measure the positive impact of predicting tidy and readable URI node patterns on RESTful Services/APIs understandability and discovery.

ACKNOWLEDGEMENTS

This research was supported by the Deanship of Research and Studies at Zarqa University/Jordan. We thank our colleagues from the Information Technologies college at Zarqa University. who provided insight and expertise that greatly assisted the research. We thank the helpful comments and suggestions of the reviewers, which have improved the research.

REFERENCES

[1] F. Palma, J. Dubois, N. Moha, and Y.-G. Guéhéneuc, "Detection of REST Patterns and Antipatterns: A Heuristics-Based Approach," pp. 230–244, 2014, doi: 10.1007/978-3-662-45391-9_16.
[2] F. Alshraiedeh and N. Katuk, "SOAP and RESTful web service anti-patterns: A scoping review," *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 8, no. 5, pp. 1831–1840, 2019, doi: 10.30534/ijatcse/2019/05852019.
[3] F. Haupt, F. Leymann, A. Scherer, and K. Vukojevic-Haupt, "A Framework for the Structural Analysis of REST APIs Florian," *Proceedings - 2017 IEEE International Conference on Software Architecture, ICSA 2017*, pp. 55–58, 2017, doi: 10.1109/ICSA.2017.40.
[4] F. Sabir, Y. G. Gueheneuc, F. Palma, N. Moha, G. Rasool, and H. Akhtar, "A Mixed-method Approach to Recommend Corrections and Correct REST Antipatterns," *IEEE Transactions on Software Engineering*, no. November, 2021, doi: 10.1109/TSE.2021.3117023.
[5] F. S. Alshraiedeh and N. Katuk, "AURI parsing technique and algorithm for anti-pattern detection in RESTful Web services," *International Journal of Web Information Systems*, vol. 17, no. 1, pp. 1–17, 2020, doi: 10.1108/IJWIS-08-2020-0052.
[6] F. Alshraiedeh, F. Alshraiedeh, N. Katuk, and M. Education, "SOAP and RESTful web service anti-patterns: A scoping review," *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 8, no. 5, 2019.
[7] A. Neagu and F. Palma, "A Systematic Literature Review on the Methodologies for Detecting REST Antipatterns in RESTful APIs," 2022.
[8] A. Arcuri, "RESTful API automated test case generation," *Proceedings - 2017 IEEE International Conference on Software Quality, Reliability and Security, QRS 2017*, pp. 9–20, 2017, doi: 10.1109/QRS.2017.11.
[9] J. R. V. Dantas and P. P. M. Farias, "An architecture for RESTful web service discovery using semantic interfaces," *Int J Semant Web Inf Syst*, vol. 16, no. 1, pp. 1–24, 2020, doi: 10.4018/IJSWIS.2020010101.
[10] S. B. A. Ben Lamine, H. Baazaoui Zghal, M. Mrissa, and C. Ghedira Guegan, "An ontology-based approach for personalized RESTful Web service discovery," *Procedia Comput Sci*, vol. 112, pp. 2127–2136, 2017, doi: 10.1016/j.procs.2017.08.235.

[11] E. Aghajani, C. Nagy, G. Bavota, and M. Lanza, "A Large-scale empirical study on linguistic antipatterns affecting apis," *Proceedings - 2018 IEEE International Conference on Software Maintenance and Evolution, ICSME 2018*, pp. 25–35, 2018, doi: 10.1109/ICSME.2018.00012.

[12] P. K. Ojha, A. Ismail, and K. Kundumani Srinivasan, "Perusal of readability with focus on web content understandability," Jan. 01, 2021, *King Saud bin Abdulaziz University*. doi: 10.1016/j.jksuci.2018.03.007.

[13] S. Ghosh, S. V. Gothe, C. Sanchi, and B. R. K. Raja, "EdATLAS: An Efficient Disambiguation Algorithm for Texting in Languages with Abugida Scripts," *Proceedings - 2021 IEEE 15th International Conference on Semantic Computing, ICSC 2021*, pp. 325–332, 2021, doi: 10.1109/ICSC50631.2021.00061.

[14] J. K. Djoko, F. Ebobisse, A. T. McBride, and B. D. Reddy, "A discontinuous Galerkin formulation for classical and gradient plasticity. Part 2: Algorithms and numerical analysis," *Comput Methods Appl Mech Eng*, vol. 197, no. 1–4, pp. 1–21, Dec. 2007, doi: 10.1016/j.cma.2007.06.027.

[15] A. De Renzis, M. Garriga, A. Flores, A. Cechich, C. Mateos, and A. Zunino, "A domain independent readability metric for web service descriptions," *Comput Stand Interfaces*, vol. 50, pp. 124–141, 2017, doi: 10.1016/j.csi.2016.09.005.

[16] J. P. J. Kincaid *et al.*, "Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel (No. RBR-8-75). Naval Technical Training Command Millington TN Research Branch.," *Naval Technical Training Command Millington TN Research Branch*, 1975, [Online]. Available: http://oai.dtic.mil/oai/oai?verb=getRecord&metadataPrefix=html&identifier=ADA006655

[17] G. Thomas, R. Derald Hartley, and J. Peter Kincaid, "TEST-RETEST AND INTER-ANALYST RELIABILITY OF THE AUTOMATED READABILITY INDEX, FLESCH READING EASE SCORE, AND THE FOG COUNT."

[18] D. Eleyan, A. Othman, and A. Eleyan, "Enhancing software comments readability using flesch reading ease score," *Information (Switzerland)*, vol. 11, no. 9, Sep. 2020, doi: 10.3390/INFO11090430.

[19] F. Zhang, Q. Zeng, H. Duan, and C. Liu, "Composition context-based web services similarity measure," *IEEE Access*, vol. 7, pp. 65195–65206, 2019, doi: 10.1109/ACCESS.2019.2915371.

[20] S. Zhou, H. Jeong, and P. A. Green, "How consistent are the best-known readability equations in estimating the readability of design standards?," *IEEE Trans Prof Commun*, vol. 60, no. 1, pp. 97–111, 2017, doi: 10.1109/TPC.2016.2635720.

[21] M. Hirsch, A. Rodriguez, J. M. Rodriguez, C. Mateos, and A. Zunino, "Spotting and Removing WSDL Anti-pattern Root Causes in Code-first Web Services Using NLP Techniques: A Thorough Validation of Impact on Service Discoverability," *Comput Stand Interfaces*, vol. 56, pp. 116–133, 2018, doi: 10.1016/j.csi.2017.09.010.

[22] Y. Zhang, N. Lin, and S. Jiang, "A Study on Syntactic Complexity and Text Readability of ASEAN English News," *Proceedings of the 2019 International Conference on Asian Language Processing, IALP 2019*, pp. 313–318, 2019, doi: 10.1109/IALP48816.2019.9037695.

[23] M. Zhang and A. Arcuri, "Open Problems in Fuzzing RESTful APIs: A Comparison of Tools," 2022.

[24] H. Antunes and C. T. Lopes, "Readability of web content," *2019 14th Iberian Conference on Information Systems and Technologies (CISTI). IEEE*, no. June, pp. 1–4, 2019, doi: 10.23919/cisti.2019.8760889.

[25] S. Zhou, H. Jeong, and P. A. Green, "How consistent are the best-known readability equations in estimating the readability of design standards?," *IEEE Trans Prof Commun*, vol. 60, no. 1, pp. 97–111, 2017, doi: 10.1109/TPC.2016.2635720.

[26] S. J. Cook, "Taxonomy of Software Readability Changes," 2022, [Online]. Available: https://scholarworks.rit.edu/theses

[27] D. Bilal, "Comparing google's readability of search results to the flesch readability formulae: A preliminary analysis on children's search queries," *Proceedings of the ASIST Annual Meeting*, vol. 50, no. 1, pp. 1–9, 2013, doi: 10.1002/meet.14505001094.