

A visual framework for software requirements traceability

Abdulkadir Ahmad Madaki^{1,2}, Wan Mohd Nazmee Wan Zainon¹

¹School of Computer Sciences, Universiti Sains Malaysia (USM), Penang, Malaysia

²Department of Mathematics and Computer Sciences, Sule Lamido University, Kafin Hausa, Nigeria

Article Info

Article history:

Received Oct 17, 2021

Revised Dec 27, 2021

Accepted Jan 13, 2022

Keywords:

Data visualization

Ease of use & usefulness

Requirement traceability

Traceability visualization

Visual framework

ABSTRACT

Requirement traceability supports several activities of software development processes such as impact analysis, requirement changes, maintenance, verification, and validation of a software system. For its effective use in those activities, the graphical representation of traceability data plays an important role. However, several traceability tools lack an excellent visual representation to present these type of data. Therefore, this paper presents a visual framework which has been designed and proposed as a prototype tool that can visualize traceability data. The framework applies data visualization techniques to represent requirements and its artefacts relationships as colour-coded symbols on a node-link diagram; users can traverse the graph with an impact analysis method to understand data and make decisions. The evaluation result shows that the proposed tool is useful and easy enough in terms of improving user interaction and to better understand requirement traceability data.

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



Corresponding Author:

Wan Mohd Nazmee Wan Zainon

School of Computer Sciences, Universiti Sains Malaysia (USM)

11800, USM, Penang, Malaysia

Email: nazmee@usm.my

1. INTRODUCTION

Requirement traceability is the process of identifying and following the life of requirements in forwards or backwards direction created during a software development lifecycle [1]. Visualization is an important technique used to represent traceability relationships between requirements and artefacts to support various software development activities such as impact analysis, requirement changes, validation of the system, maintenance, and software evolution [2]. During software development, a large number of complex traceability relationships are generated [3]. Several traceability tools lack excellent visual representation to trace relationships between requirements and artefacts [4]. Therefore, there is a need to have a graphical visual tool that can help to visualize and represent traceability relationships between requirements and artefacts [5]. With the help of visualization techniques and graphical notations, the user can easily visualize tracing the relationships between requirements and artefacts [6]-[8].

In this paper, we proposed a visual framework in order to visualize the traceability relationships between requirements and its artefacts during the software development process or requirement change. We adopt the graphical user interface design for visual information-seeking mantra, “overview first, zoom and filter, then details-on-demand”, as proposed by Schneiderman [9]. The tool uses graphical notations to represent the traceability of requirements and artefacts relationships bi-directionally. Users can traverse the graph for impact analysis and reduce visual clutter to understand their data quickly and make data visualization has been recognized as a significant research field focused on the application of visualization techniques to help people understand and analyze traceability data [10]. Visualization techniques, design principles, human perception, and colour theory are applied in data presentation [9]. Traceability techniques

such as graphs, matrix, and hyperlinks are used to show complex data [11]. A graph-based visualization is the most common visualization technique used by traceability systems [12], [13]. It supports the visualization of an overview of traceability links between multiple requirements and artefacts as nodes, with relationships as edges. Li and Maalej [7] recently examined and compared graphs, matrix, lists, and hyperlinks techniques in an empirical study to show which of these visualization techniques are more suitable to be conducted in any given context. The study highlights the traceability graph as the best to support management tasks. Several visualization tools were developed to visualize the traceability relationships between requirements and artefacts as a node-link diagram representation on a network graph [14]-[17].

Rubasinghe *et al.* [5] proposed a traceability support visualization tool for heterogeneous artefacts in DevOps environments. The tool eliminates and updates conflicts raised among artefacts, while the developed artefacts in the project are outdated. They proposed traceability process model consists of artefact change detection, change impact analysis, and change propagation. Moreover, this tool provides multi-user accessibility and is integrated with a prominent DevOps tool stack to enable collaborations. Whenever the number of artefacts increased, the graph complexity of viewing the relationships also increased. However, with the predefined overcrowded relationships in the graph view, the graph complexity was reduced to provide an effective analysis vision. In contrast, visualization of the links may provide the release of each software product. Figure 1 shows an example of an interactive traceability graph, where nodes denote the artefact and edges show the links.

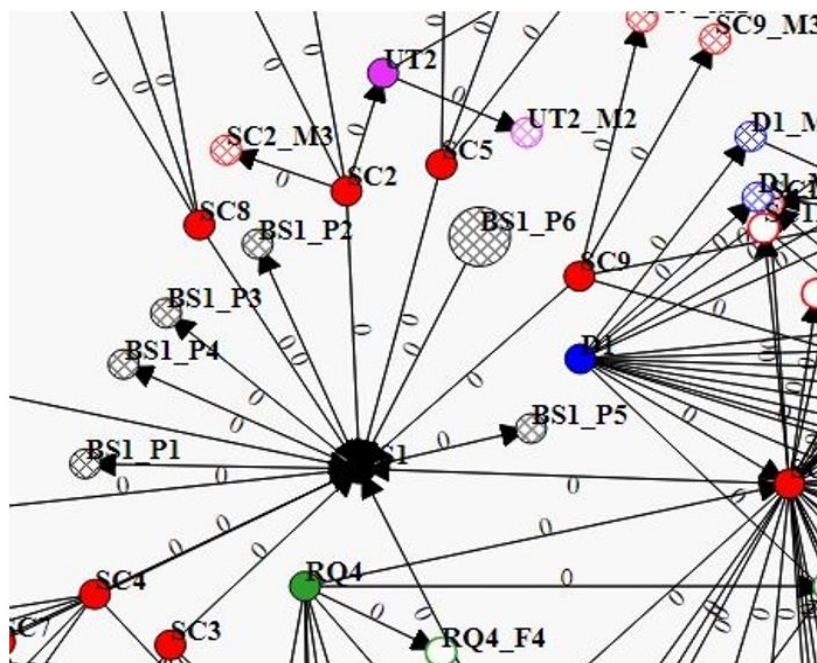


Figure 1. An example of the change propagated for interactive traceability graph [5]

Similarly, another study by Beier and Muller [3] presents a prototype visualization tool for traceability artefacts called "Ariadne's eye". It is an interactive method of scaling, zooming, and folding in (node-link) of a vertical tree and balloon tree layouts. The tool is visualized to compare three geometrical forms, namely circle, eclipse, and polygon, that help users understand and select a type of hierarchical framework to link all elements and their direct ancestral line easily. The systematic visualization layout for complex graphs consists of numerous hierarchical artefacts. The spatially separates the hierarchical artefacts and traces the relationships to make it readable and understandable to the users on the graph, as shown in Figure 2. The study is limited and valid with two issues; the moderately low number of participants and the number of elements within an artefact far away from the industrial artefacts. They further refined and advanced the layout according to user feedback. They implemented their final ideas as an interactive visualization tool called Ariadne's eye. The result analysis method is displayed where the mean values per task and layout are given.

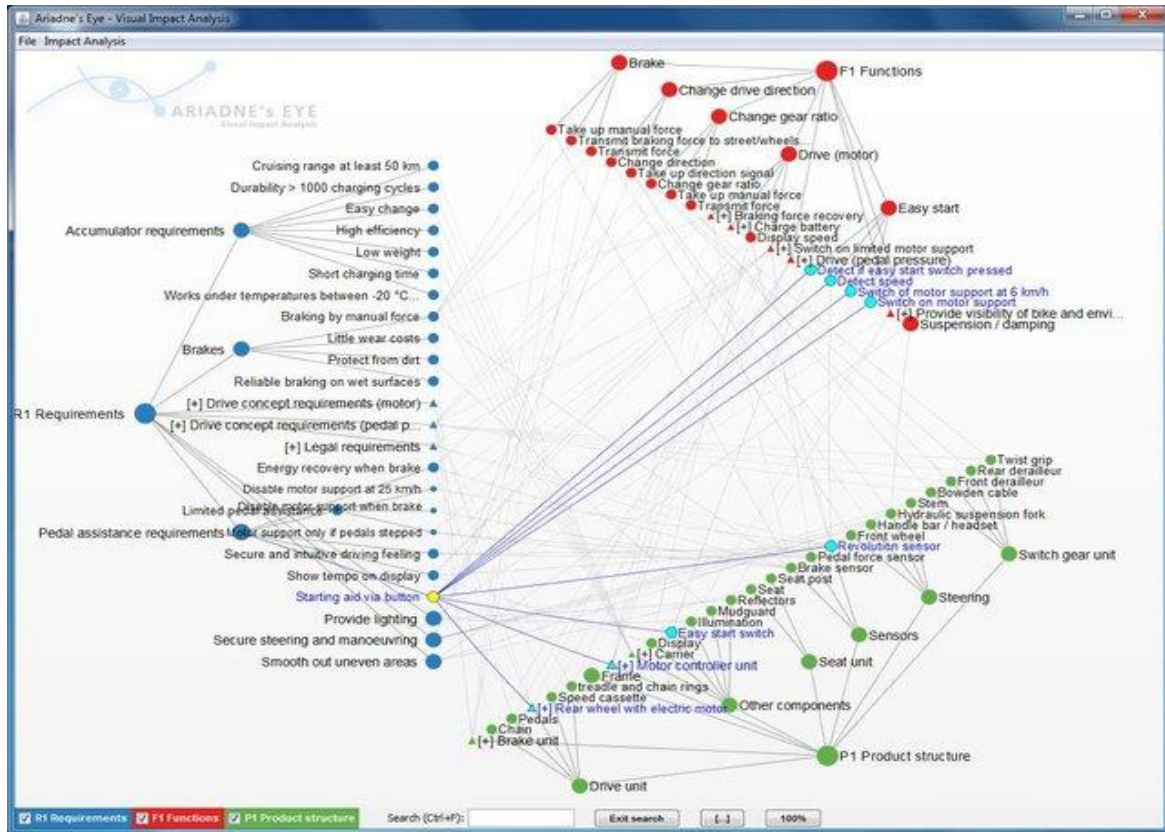


Figure 2. Ariadne's eye visualization tool [3]

2. PROPOSED METHOD

This research study developed a visual framework design to help users visualize requirement traceability data. The visual framework follows the design of advanced graphical user interface (GUI) guide for visual information seeking mantra, “overview first, zoom and filter, then details on-demand”, which was proposed by Schneiderman in 1996. The regular use of the mantra is an indication that shows the excellent design and helps many practitioners in different design scenarios [18]. In Figure 3, the overview of the proposed visual framework is presented.

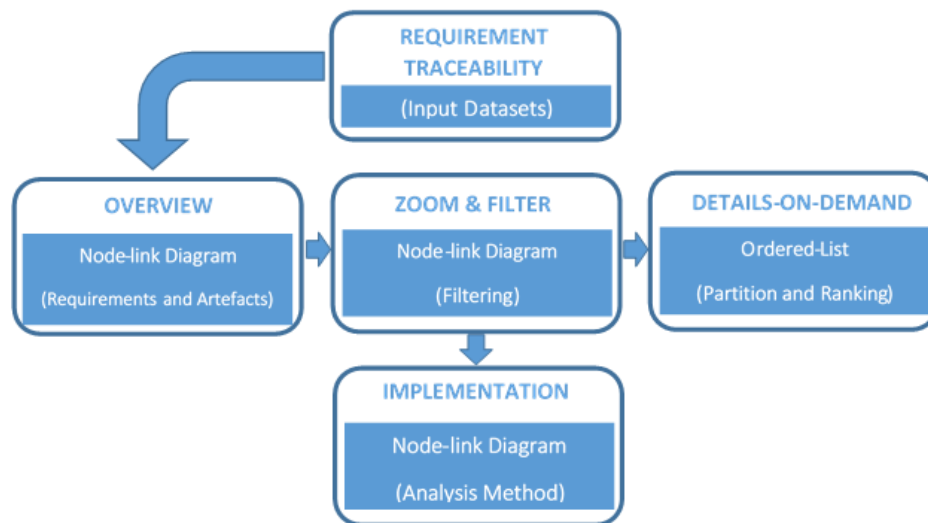


Figure 3. The proposed visual framework

In Figure 3, it can be observed that the visual framework provides data input, visualization approaches, and implementation of the prototype tool called VizTraceArtefacts. The use of visualization techniques helps to display the overview of traceability data and enables the user to modify the detailed information displayed. The selected visual cluster can be filtered to reduce the number of visual clutters for requirements and artefacts relationships on the network graph.

Moreover, the tool enables the analysis of understanding traceability data in a different view. Figure 4 shows some typical shapes and colours used to represent demographic appearances of requirements, artefacts, relationships, and directions. The colour-coded shapes represent requirements and artefacts using cycles, while the colour-coded arrows represent relationships and directions, all depending on the type of colour used.

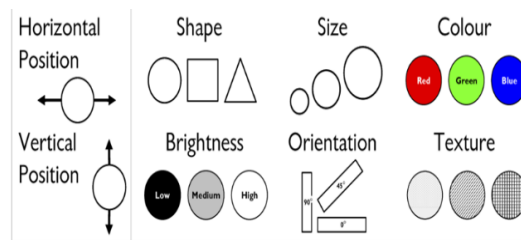
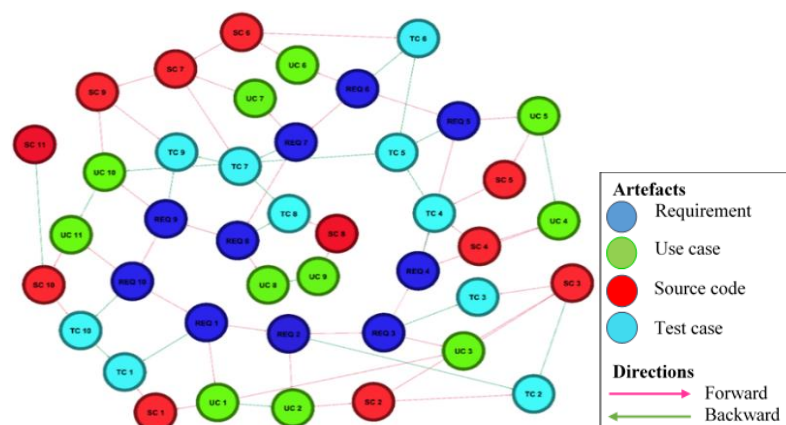


Figure 4. Example of the types of color-coded symbols used

The implemented prototype tool was made through the application of an adopted novel methodological guide for visual information seeking mantra [18]. The methodology is presented and categorized into three (3) phases that include preliminary study and analysis, designing the framework and its implementation, and evaluation. The selection of visualization techniques to implement the tool follows the design of advanced GUI guide for visual information seeking mantra, “overview first, zoom and filter, then details on-demand”, as proposed by Schneiderman in 1996. The frequent use of the mantra is evidence in which many practitioners find it helpful in different design scenarios [18], [19].

The first technique applied is called an “overview first”; it is a perspective method of an understanding dataset with a complete diagram structure familiar with data designs [20]. This technique represents the overview of data as a node-link diagram to improve user understandability and communication [21]. “Node-link diagram” is a type of visualization on a graph that shows relationships and represents their connections as nodes and edges to identify the type of relationships that exist between a group of entities [22]. Figure 5 shows an overview of our traceability visualization from VizTraceArtefacts. The nodes represent requirements and artefacts; the links represent relationships and show directions of forward and backward. All the links are coloured based on the type in order to identify them on the network.



represent data easily on the network” [18]. The “zoom” technique is a process of fine-tuning to adjust components of data position and size on the screen, while “filter” is considered as alteration and navigation to represent an essential viewpoint on the diagram [23]. “Zooming and filtering” allow users to filter nodes based on their needs because many visualized components are limited, therefore the perceptual clarity of users needs an improvement. Figure 6 shows the VizTraceArtefacts zoom and filter traceability visualization.

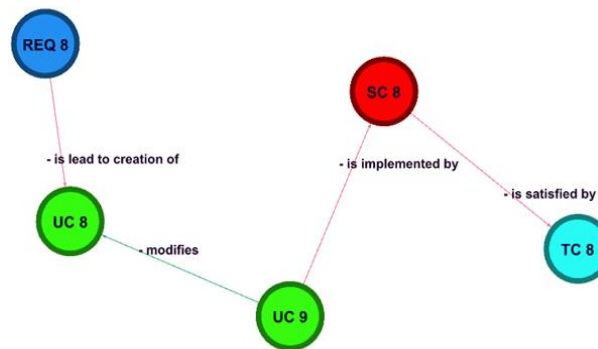


Figure 6. Zoom and filter visualization showing relationships between a requirement and artefacts

The last technique applied is “details-on-demand” to offer more information without altering the design view. Therefore, the “ordered-lists” method is used to enable a user to find outliers on the network. Accordingly, “ranking” is a general approach to organize or disorganize a set of items by calculating the rank of an individual item according to the value of one to many of its attributes [24]. This approach enables a user to combine the attribute ranking and partition, ordered-lists and colour-coding to measure the scale of nodes and edges on the network. Furthermore, it also assists in the evaluation of nodes performance that has relation with one another and prioritizes them according to the ranking. Table 1 shows the ordered list for comparing rankings.

Table 1. Ordered list for comparing rankings

Rank	Node (artefacts)	Relationship type
4.00	Requirements (REQ)	Is tested by
3.00	Use case (UC)	Depends on
3.00	Source code (SC)	Is verified on
2.00	Test case (TC)	Influenced
2.00		Is implemented by
1.00		Lead to creation of
1.00		Is modified by
1.00		Is satisfied by

We used colour-coded circles for each type of node to support usability aspects in the establishment of traceability artefacts and relationships. We considered the requirement-use case, use case-source code, source code-test case requirement-test case, and use case-test case artefacts. However, the design was altered to directly reflect in artefacts; consequently, the nodes represent requirements and artefacts while the edges show the traceability relationships and directions. As shown in Figure 5, the nodes in blue represent requirements, the green nodes represent use cases, the red nodes represent source codes, and the light blue nodes represent test cases. The colour arrows represent links that show the relationships with directions between requirements and artefacts. ForceAtlas2 layout algorithm is used to position whole nodes on the network. It is a “spring analogue method of drawing graphs to position nodes and links that are iteratively refined and present them until the whole energy of the system is minimized” [25]. Therefore, “node-link diagrams significantly leave an empty background space and come across scalability issues when applied to larger graphs” [22]. The ranking method used to measure the number of connected nodes is the “degree” between requirements and artefacts. Each node is ranked based on the method of an ordered list, and a partition is allocated to a range of colour groups. The users can traverse the graph for impact analysis to further understand the data and make decisions.

3. EVALUATION

In this study, a case study was selected to evaluate the proposed tool because of its usefulness in terms of examining how “scientific theories and models work in the real world”. Hence, it is an appropriate way to conduct an evaluation and procedure to evaluate the effectiveness of the visual framework and the proposed tool. Subject and relevance are the first substances and groundwork of the case study by means of using a small set of participants that produce candid and convincing stories from real users that interact with the visualization tool without demanding more time [23]. Therefore, a total of thirty-five (35) respondents participated in this case study with profiles like the target users. The data collection was performed at the School of Computer Sciences, Universiti Sains Malaysia in Pulau Pinang, Malaysia (Main Campus).

The perceived usefulness is “a point that person trusts as using a certain system is skillful of an existence that is very useful.” Thus, a positively perceived usefulness tool is a lively performance that a user believes of its existence. On the other hand, the perceived ease of use is “a point that person trusts as using a certain system will highly reduce effort”. Therefore, the ease of use tool will be accepted by users and more likely to be acknowledged.

The usefulness and ease of use are the most significant user bases that measure the usability features of a traceability tool and general concepts of usability. All the users were required to answer both usefulness and ease of use questions in this case study according to their views and observations made on VizTraceArtefacts. The questions on usefulness and ease of use are based on research on the self-efficiency theory, research on the cost-benefit paradigm from behavioural decision theory, and research on adopting innovations [25].

The evaluation was carried out to determine the usability of the prototype tool (usefulness) and (ease of use) to improve user interaction and understanding on requirement traceability. Therefore, the questionnaire is categorized into three sections; in the first section (1), it consists of demographic questions to gather respondents’ information. The second section (2) consists of assessment questions provided for respondents to answer according to their understanding by using VizTraceArtefacts. The third section (3) also consists of some questions to evaluate the proposed visual framework according to the two (2) selected variables, i.e., perceived usefulness and perceived ease of use. All the respondents who interacted with VizTraceArtefacts used the same type of questionnaire.

Finally, an open-ended evaluation approach adapted from [26] was used to evaluate the effectiveness of the visualization techniques applied for this proposed tool. Therefore, the participants were given a questionnaire and room space to fill up and elaborate on the usefulness and effectiveness of the visualization techniques and statistical analysis method applied to the visual framework. Statistical and network analysis techniques were used to evaluate the accuracy of the overall traceability graph that consists of 42 nodes and 66 edges. Degree and eigenvector centrality measures were applied to identify the most influential and important artefacts on the graph.

Figure 7 shows the eigenvector centrality of the traceability links over all artefact nodes that states the average node importance in the traceability graph based on the average traceability connection of each node. The average change is calculated as 0.0091 using Gephi. There are more noninfluential nodes, as a maximum of 42 nodes have a score of 0 eigenvector centrality. Two nodes with the highest eigenvalue centrality of the value 1 show the most influential nodes. According to the result, the relative eigenvector centrality (EVC) measures with high values are likely to be a high importance artefact. This confirms the accuracy of traceability links between the most influential artefacts and indicates the least influential artefacts with very low centrality values.

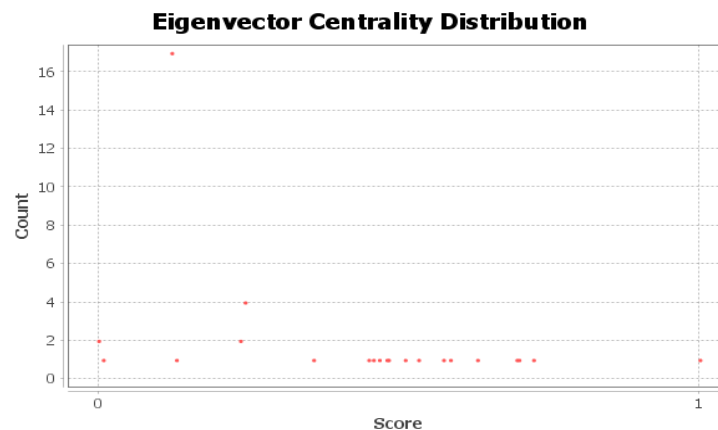


Figure 7. Eigenvector centrality distribution in overall traceability

4. RESULTS AND DISCUSSION

The findings of this study show that the proposed tool is found to be very useful and easy in terms of visualizing requirement traceability data by all respondents. The study findings on usefulness of this framework show that it is effective in helping users to improve their understanding and make decisions. Whenever the complexity of artefacts relationships increases, visualization gives users an impression to visualize, trace and filter artefacts to discover information easily. The result is not statistically significant as the case study has a small number of participants. Therefore, there is only a minor difference between participants in terms of speed and time taken to complete the assessment.

According to the bar chart, all recorded items have the highest number of mean scores that is between "4 to 5". All the respondents have assured that the proposed tool make their job easier and they clearly understand traceability data better. The use of zoom and filter techniques help display the necessary information and reduced visual clutter that increased the effectiveness of the tool (mean of 4.76). Some respondents acknowledged the tool as it reduces searching effort and clicks features, which makes work very useful and more comfortable (mean of 4.16) for simplifying their work. As a result of increasing job performance (mean of 4.37) and reduction of time taken to complete a task, this enables them to perform their job easier (mean of 4.0). Moreover, the increase in productivity items is slight as an opinion made by a single respondent (mean of 4.3) due to task complexity and confusion at the time to join more than one visualization technique. Figure 8(a) shows the score mean of usefulness and Figure 8(b) shows the score for ease of use between all metrics.

The result of ease of use shows that the proposed tool is easy and turns out to be skillful, clear, and understandable as all respondents answered all the questions appropriately. In the recorded items, clear and understandable (mean of 4.8), skillful (mean of 4.6), remember (mean of 4.0), easy to learn, and easy to use (mean of 4.3) achieved the highest score mean for ease of use factors. On the other hand, controllable achieved the lowest mean score of (3.5) as a small number of respondents confirmed that they faced problems in navigating to each visualization technique to help them perform their tasks. Thus, in the case of a first-time user, it is impossible to learn something new quickly in a short time. Due to this issue, a manual with some guidelines from the researcher was provided to simplify respondents' learning process. Moreover, an open-ended evaluation result shows a perfect meaningful answer from the respondents' understanding after using the proposed tool which inspired this discussion towards the use of an open-ended question. The respondents were all satisfied with how each visualization technique applied to the visual framework helps them improve their understanding and complete the assessment tasks.

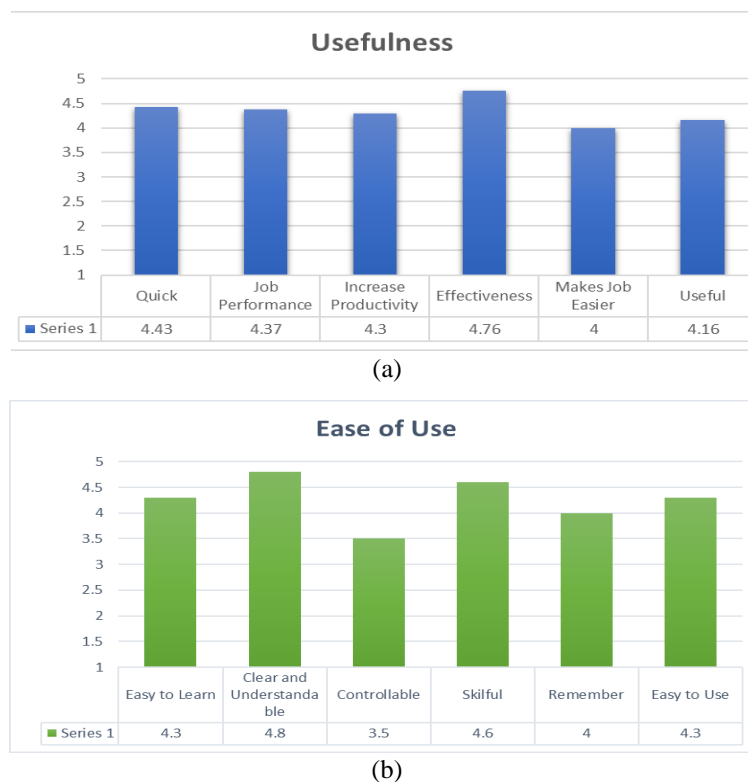


Figure 8. The mean score of (a) usefulness and (b) ease of use

5. CONCLUSION

In this paper, we proposed a visual framework and a visualization tool called VizTraceArtefacts that supports the visualization of software requirement traceability data. The tool applies data visualization techniques to represent requirements traceability data as colour-coded symbols on a node-link diagram. The main contribution of this framework is that it can help users to improve their understanding and interaction, develop products rapidly, increase software quality, reduce maintenance effort, and manage requirement change frequently during the software development lifecycle. The evaluation result obtained from usability measures in the considered case study shows that the visual framework is significant enough in terms of improving user interaction and understanding. The tool can be further enhanced with a filtering search mechanism technique due to the efficiency benefits to search for requirements using keywords and limit the search filter to only specific relationships, directions, link number, and properties.




REFERENCES

- [1] P. Lago, H. Muccini, and H. van Vliet, "A scoped approach to traceability management," *Journal of Systems and Software*, vol. 82, no. 1, pp. 168-182, January 2009, doi: 10.1016/j.jss.2008.08.026.
- [2] G. Spanoudakis and A. Zisman, "Software traceability: a roadmap," in *Handbook of Software Engineering and Knowledge Engineering: Recent Advances*, World Scientific, vol. 3, pp. 395-428, 2005, doi:10.1142/9789812775245_0014.
- [3] V. Dashuber, P. Philippsen, and J. Weigend, "A Layered Software City for Dependency Visualization," in *Proceedings of the 16th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2021): IVAPP*, 2021, vol. 3, pp. 15-26, doi: 10.5220/0010180200150026.
- [4] A. Rodrigues, M. Lencastre, and G. A. De A. C. Filho, "Multi-VisioTrace: Traceability Visualization Tool," *2016 10th International Conference on the Quality of Information and Communications Technology (QUATIC)*, 2016, pp. 61-66, doi: 10.1109/QUATIC.2016.019.
- [5] I. Rubasinghe, D. Meedeniya, and I. Perera, "SAT-Analyser Traceability Management Tool Support for DevOps," *Journal of Information Processing Systems*, vol. 17, no. 5, pp. 972-988, 2021, doi:10.3745/JIPS.04.0225.
- [6] S. Altaf, A. Shah, N. Imtiaz, A. S. Shah, and S. F. Ahmed, "Visualization representing benefits of pre-requirement specification traceability," *International Journal of Engineering & Technology*, vol. 7, no. 2, pp. 44-52, 2018, doi: 10.14419/ijet.v7i2.5.10052.
- [7] Y. Li and W. Maalej, "Which traceability visualization is suitable in this context? a comparative study," in *International Working Conference on Requirements Engineering: Foundation for Software Quality*, vol. 7195, pp. 194-210, 2012, doi: 10.1007/978-3-642-28714-5_17.
- [8] D. Meedeniya, I. Rubasinghe, and I. Perera, "Software artefacts consistency management towards continuous integration: a roadmap," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 4, pp. 100-110, 2019, doi: 10.14569/IJACSA.2019.0100411.
- [9] B. Shneiderman, "The eyes have it: A task by data type taxonomy for information visualizations," in *Proceedings 1996 IEEE symposium on visual languages*, 1996, pp. 336-343, doi: 10.1109/VL.1996.545307.
- [10] P. Federico, M. Wagner, A. Rind, A. Amor-Amorós, S. Miksch, and W. Aigner, "The Role of Explicit Knowledge: A Conceptual Model of Knowledge-Assisted Visual Analytics," *2017 IEEE Conference on Visual Analytics Science and Technology (VAST)*, 2017, pp. 92-103, doi: 10.1109/VAST.2017.8585498.
- [11] D. Sacha, H. Senaratne, B. C. Kwon, G. Ellis, and D. A. Keim, "The Role of Uncertainty, Awareness, and Trust in Visual Analytics," in *IEEE Transactions on Visualization and Computer Graphics*, vol. 22, no. 1, pp. 240-249, 31 Jan. 2016, doi: 10.1109/TVCG.2015.2467591.
- [12] D. Meedeniya, I. Rubasinghe, and I. Perera, "Artefact consistency management in DevOps practice: A survey," in *Tools and Techniques for Software Development in Large Organizations: Emerging Research and Opportunities*. Hershey, PA: IGI Global, 2020, pp. 98-129, doi: 10.4018/978-1-7998-1863-2.ch005.
- [13] S. C. Jayaraman and M. Anand, "The impact of visualizing traceability links for evolving requirements in software maintenance-A controlled experiment," *International Journal of Intelligent Engineering and Systems*, vol. 10, no. 3, pp. 235-244, 2017, doi: 10.22266/ijies2017.0630.34.
- [14] X. Chen, J. Hosking, and J. Grundy, "Visualizing traceability links between source code and documentation," *2012 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*, 2012, pp. 119-126, doi: 10.1109/VLHCC.2012.6344496.
- [15] K. Kamalabalan *et al.*, "Tool support for traceability of software artefacts," *2015 Moratuwa Engineering Research Conference (MERCon)*, 2015, pp. 318-323, doi: 10.1109/MERCon.2015.7112366.
- [16] X. Chen, "Extraction and visualization of traceability relationships between documents and source code," *ASE '10: Proceedings of the IEEE/ACM International Conference on Automated Software Engineering*, September 2010, pp. 505-509, doi: 10.1145/1858996.1859098.
- [17] W.M.N. Zainon, A.Z. Talib, B. Belaton, A New Framework for Phylogenetic Tree Visualization. In: Zaman H.B. et al. (eds) *Visual Informatics: Sustaining Research and Innovations*. IVIC 2011. Lecture Notes in Computer Science, vol 7067. Springer, Berlin, Heidelberg, 2011, doi: 10.1007/978-3-642-25200-6_33.
- [18] B. Craft and P. Cairns, "Beyond guidelines: what can we learn from the visual information seeking mantra?," Ninth International Conference on Information Visualisation (IV'05), 2005, pp. 110-118, doi: 10.1109/IV.2005.28.
- [19] J. R. Lewis, "Psychometric evaluation of an after-scenario questionnaire for computer usability studies: the ASQ," *ACM Sigchi Bulletin*, vol. 23, no. 1, pp. 78-81, 1991, doi: 10.1145/122672.122692.
- [20] F. Tian, T. Wang, P. Liang, C. Wang, A. A. Khan, and M. A. Babar, "The Impact of Traceability on Software Maintenance and Evolution: A Mapping Study," *Journal of Software: Evolution and Process*, vol. 33, no. 10, p. e2374, 2021, doi: 10.1002/smr.2374.
- [21] T. von Landesberger, S. Bremm, N. Andrienko, G. Andrienko, and M. Tekušová, "Visual analytics methods for categoric spatio-temporal data," *2012 IEEE Conference on Visual Analytics Science and Technology (VAST)*, 2012, pp. 183-192, doi: 10.1109/VAST.2012.6400553.




- [22] B. Shneiderman and C. Plaisant, "Strategies for evaluating information visualization tools: multi-dimensional in-depth long-term case studies," in *Proceedings of the 2006 AVI workshop on BEyond time and errors: novel evaluation methods for information visualization*, May 2006, pp. 1-7. doi:10.1145/1168149.1168158.
- [23] S. Gratzl, A. Lex, N. Gehlenborg, H. Pfister, and M. Streit, "LineUp: Visual Analysis of Multi-Attribute Rankings," in *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 12, pp. 2277-2286, December 2013, doi: 10.1109/TVCG.2013.173.
- [24] E. H. Chi, "A taxonomy of visualization techniques using the data state reference model," *IEEE Symposium on Information Visualization 2000. INFOVIS 2000. Proceedings*, 2000, pp. 69-75, doi: 10.1109/INFVIS.2000.885092.
- [25] F. D. Davis, "Perceived usefulness, perceived ease of use, and user acceptance of information technology," *MIS Q.*, pp. 319-340, 1989, doi: 10.2307/249008.
- [26] L. K. Long, L. C. Hui, G. Y. Fook, and W. M. N. W. Zainon, "A study on the effectiveness of tree-maps as tree visualization techniques," *Procedia Computer Science*, vol. 124, pp. 108-115, 2017, doi: 10.1016/j.procs.2017.12.136.

BIOGRAPHIES OF AUTHORS



Abdulkadir Ahmad Madaki    received Bachelor of Science (Second Class Upper Hons.) in Software Engineering from University of East London in 2013, and currently pursuing his Master's Degree in Computer Science Universiti Sains Malaysia. Currently, he is a graduate assistant in department of Computer Science, Sule Lamido University. His research interests are primarily in the area of requirement engineering, data visualization and software analysis and design patterns, where he is the author/co-author of over 3 research publications. He can be contacted at email: engr.ahmadmadaki@gmail.com.



Wan Mohd Nazmee Wan Zainon    received the Ph.D degree in computer science from Universiti Sains Malaysia. He is currently an Associate Professor at the School of Computer Sciences, Universiti Sains Malaysia. His research interests include visual computing, data mining and software engineering with a focus on software reuse, requirement engineering practices, visual data mining, and big data analytics. He can be contacted at email: nazmee@usm.my.