IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 11, Issue 10, October 2023

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.379

9940 572 462 | 6381 907 438 | ijircce@gmail.com | www.ijircce.com

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 || A Monthly Peer Reviewed & Referred Journal |

|| Volume 11, Issue 10, October 2023 ||

| DOI: 10.15680/IJIRCCE.2023.1110040 |

Multi-Criteria Framework for the Selecting Software Engineering Tools

Rishi Mewara, Prof. Minal Toley

Dept. of Computer Engineering, Ajeenkya DY Patil School of Engineering, Lohegaon, Maharashtra, India

ABSTRACT: Selecting the right software engineering tools is a critical decision for organizations aiming to develop high-quality software products efficiently. This decision is further complicated by the vast array of tools available, each catering to specific needs and requirements. To assist in this decision-making process, a multi-criteria framework is proposed in this paper. The Multi-Criteria Framework for Selecting Software Engineering Tools (MCFSSET) provides a structured approach to evaluate and choose software engineering tools based on multiple criteria. The framework encompasses a comprehensive set of evaluation criteria, which includes aspects such as functionality, cost, scalability, compatibility, user-friendliness, support, and more. Each of these criteria is assigned weights to reflect their relative importance in the specific context of the organization.

The framework utilizes various decision-making techniques, such as Analytic Hierarchy Process (AHP), Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS), and Weighted Sum Model (WSM), to rank and prioritize the available tools based on the chosen criteria. This enables organizations to make informed decisions, taking into account their unique requirements and constraints.

MCFSSET aims to streamline the tool selection process, enhance the transparency of decision-making, and ensure that software engineering tools align with the organization's strategic goals. By offering a systematic and data-driven approach, it enables software development teams to make choices that will lead to more efficient and effective development processes, ultimately resulting in higher-quality software products. This paper not only presents the framework but also provides a practical example of its application in the selection of software engineering tools, demonstrating its utility in real-world scenarios.

KEYWORDS: Software engineering tools, tool selection, multi-criteria decision-making, framework, criteria, software development, decision analysis.

I. INTRODUCTION

In the rapidly evolving field of software development, the choice of software engineering tools can significantly impact the efficiency, quality, and success of a project. Selecting the right tools is akin to assembling a toolkit for a craftsman, each tool serving a specific purpose to facilitate the creation of a masterpiece. However, the task of identifying the most suitable tools from the extensive array available is often a challenging and critical decision for software development teams and organizations.

Historically, tool selection processes have largely relied on ad-hoc decisions or recommendations from peers, leading to suboptimal choices and missed opportunities. The diversity of tools available in the market, ranging from integrated development environments (IDEs) to project management and testing tools, further complicates this decision-making process. Recognizing this challenge, this paper proposes a Multi-Criteria Framework for Selecting Software Engineering Tools (MCFSSET) that aims to bring structure, objectivity, and rigor to the tool selection process. MCFSSET is designed to assist software development teams, project managers, and decision-makers in making informed choices that align with the organization's specific needs, goals, and constraints.

The need for such a framework is underscored by the multifaceted nature of tool selection decisions. Software engineering tools are evaluated based on a multitude of criteria, which include but are not limited to functionality, cost, scalability, compatibility, user-friendliness, support, and strategic alignment. These criteria often interact with one another, making it challenging to assess and prioritize tools accurately. In this paper, we outline the development and application of the MCFSSET framework, which includes a comprehensive set of evaluation criteria, the assignment of relative weights to these criteria, and the utilization of various multi-criteria decision-making techniques. The

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 || A Monthly Peer Reviewed & Referred Journal |

|| Volume 11, Issue 10, October 2023 ||

| DOI: 10.15680/IJIRCCE.2023.1110040 |

framework offers a structured approach that enables organizations to assess and rank software engineering tools systematically, thereby aiding in the selection process.

By employing the MCFSSET framework, software development teams can make decisions that not only meet their current needs but also align with long-term objectives. The result is a more efficient, effective, and strategic approach to tool selection, ultimately contributing to the development of higher-quality software products. This paper aims to provide a foundation for understanding the importance of a structured tool selection process and lays the groundwork for the subsequent discussion of the MCFSSET framework, its components, and its practical application in real-world software development scenarios.

III. METHODOLOGY

1. Problem Definition and Scope:

- Define the problem and scope of software engineering tool selection within the organization.
- Identify the key stakeholders and their roles in the decision-making process.
- Set clear objectives for the tool selection process, considering the organization's strategic goals.

2. Criteria Identification:

- Collaboratively establish a comprehensive list of criteria that will be used to evaluate software engineering tools. Criteria may include functionality, cost, scalability, compatibility, user-friendliness, support, and others.
- Ensure that the criteria are tailored to the specific needs and constraints of the organization.

3. Criteria Weighting:

- Gather input from stakeholders to assign relative weights to each of the identified criteria. These weights reflect the importance of each criterion in the context of the organization's goals.
- Utilize techniques such as Analytic Hierarchy Process (AHP) or the Delphi method for achieving consensus on criteria weights.

4. Tool Evaluation:

- Collect information about available software engineering tools in the market.
- Evaluate each tool based on the established criteria and their respective weights.

5. Multi-Criteria Decision Making:

- Apply multi-criteria decision-making techniques to rank and prioritize the software engineering tools.
- Common techniques include Analytic Hierarchy Process (AHP), Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS), Weighted Sum Model (WSM), or the Electre method.
- Normalize and aggregate the criteria scores to obtain a consolidated evaluation for each tool.

6. Sensitivity Analysis:

- Conduct sensitivity analysis to understand how changes in criteria weights can impact tool rankings.
- Identify robust choices that remain preferred under various weight scenarios.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 || A Monthly Peer Reviewed & Referred Journal |

|| Volume 11, Issue 10, October 2023 ||

| DOI: 10.15680/IJIRCCE.2023.1110040 |

7. Selection and Documentation:

- Select the software engineering tools based on their rankings and how well they align with the organization's objectives and constraints.
- Document the rationale behind the selection, including the criteria used, their respective weights, and the final tool choices.

8. Validation and Testing:

- Implement the selected tools in a controlled environment or pilot project to ensure they meet the expected performance and integration requirements.
- Monitor their effectiveness in real-world scenarios and adjust the decision if necessary.

9. Feedback and Iteration:

- Gather feedback from users and project teams regarding the selected tools' performance.
- Continuously monitor the tools' relevance and adjust the selection if the organization's needs change over time.

10. Documentation and Reporting:

- Document the entire tool selection process, including the criteria, weights, evaluation results, and the rationale for tool choices.
- Prepare a comprehensive report for stakeholders, outlining the decision-making process and the selected tools.

11. Implementation and Training

- Facilitate the implementation of the chosen software engineering tools within the organization.
- Provide training and support to ensure that teams can effectively utilize the tools.

12. Continuous Improvement:

- Encourage a culture of continuous improvement by periodically reviewing and updating the tool selection process and the criteria used.
- Stay current with advancements in software engineering tools and adapt the framework as needed.

IV. ANALYSIS

This framework streamlines tool selection, providing a structured approach that improves the quality and efficiency of software development. It’s multi-criteria evaluation and decision making techniques enhance strategic alignment.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 || A Monthly Peer Reviewed & Referred Journal |

|| Volume 11, Issue 10, October 2023 ||

| DOI: 10.15680/IJIRCCE.2023.1110040 |

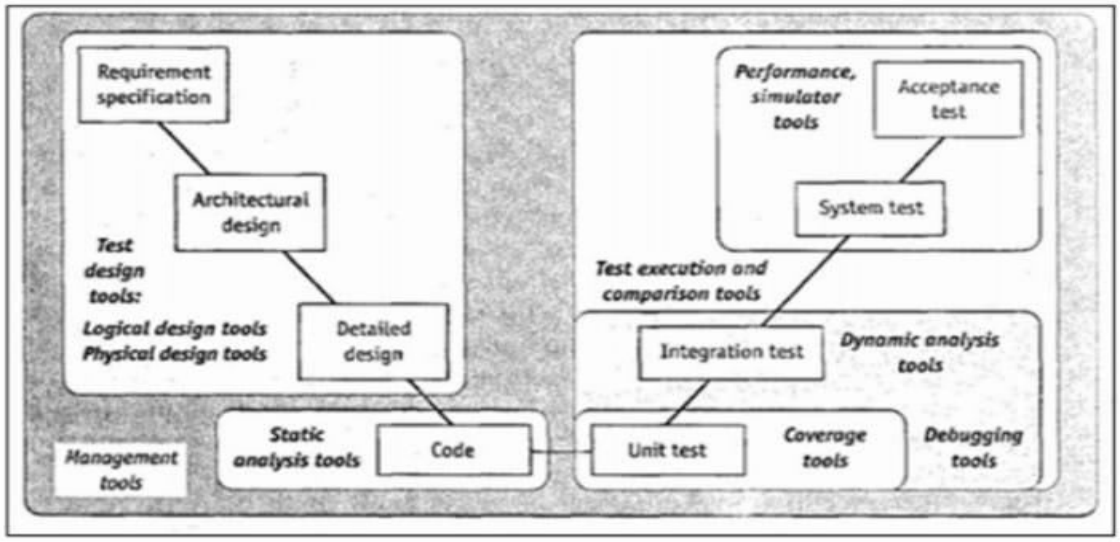

FIGURE 1. Testing tools spans the software development life cycle

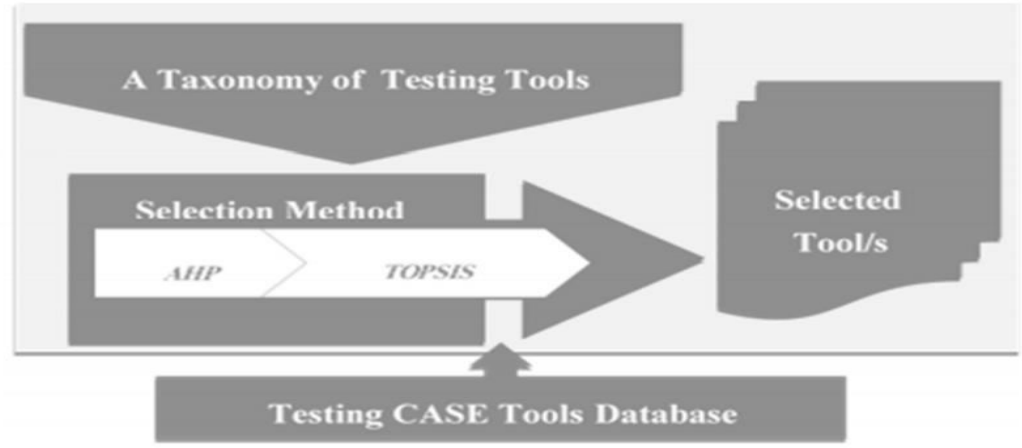

FIGURE 2. Multi-Criteria Framework for selecting testing tools

FIGURE 3. The proposed taxonomy for testing tool

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 || A Monthly Peer Reviewed & Referred Journal |

|| Volume 11, Issue 10, October 2023 ||

| DOI: 10.15680/IJIRCCE.2023.1110040 |

TABLE 1. Used Software with Criteria

	Software	Testing Criteria Based on the Proposed Taxonomy							
		Programing Language	Software Technique	Operating System/ Platform	Testing Level	Quality Assurance Requirements	Testing Technique	Tool Acquiring Method	Software testing Model /Approach
1	Accounting System UST	Oracle, SQL, PHP	Database, Web App.,	Win, Unix	Unit, Integration Testing	Security, Reliability, Usability, Efficiency, Functionality	Black-Box White-Box	Free, Rent	Waterfall
2	Human Resources System UST	Oracle, SQL, Java, PHP	Database, Web App., Mobile App.	Win, Unix, Android, iOS	Unit, Integration Testing	Functionality Security, Reliability, Usability	Black-Box White-Box	Free, Rent, Bough	Waterfall, Agile, Incremental
3	Open Education System UST	Oracle, SQL, Java, PHP	Database, Web App., Mobile App.	Win, Android, iOS	Unit, Integration Testing	Functionality , Reliability, Usability	Black-Box White-Box	Free, Rent	Agile, XP
4	Registration System UST	Oracle, SQL, PHP	Databases , Web App.	Win, Unix	Unit, Integration Testing	Functionality , Security, Reliability, Usability	Black-Box White-Box	Free, Rent	Waterfall
5	Balanced Scorecard System UST	PHP, SQL, Java Script	Database, Web App.	Win	Unit, Integration Testing	Functionality , Usability	Black-Box White-Box	Free, Rent	Agile
6	Service and Maintenance System UST	Oracle, SQL, Java Php	Database, Web App., Mobile App.	Win, Android, iOS	Unit, Integration, System Testing	Functionality , Usability	Black-Box White-Box	Free, Rent	Agile, Incremental
7	Journals Editing System UST	PHP, SQL, Java Script	Database, Web App.	Win	Unit, Integration Testing	Functionality , Reliability, Usability	Black-Box White-Box	Free, Rent	Agile, Incremental
8	University of Science and Technology Website	PHP, SQL, Java Script	Database, Web App.	Win	Unit, Integration Testing	Functionality , Usability	Black-Box White-Box	Free, Rent	Waterfall, Agile
9	Learning Management System (CLMS) UST	PHP, SQL, Java Script	Database, Web App.	Win	Unit, Integration, System Testing	Functionality , Usability	Black-Box	Free	Agile, Incremental

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 || A Monthly Peer Reviewed & Referred Journal |

|| Volume 11, Issue 10, October 2023 ||

| DOI: 10.15680/IJIRCCE.2023.1110039 |

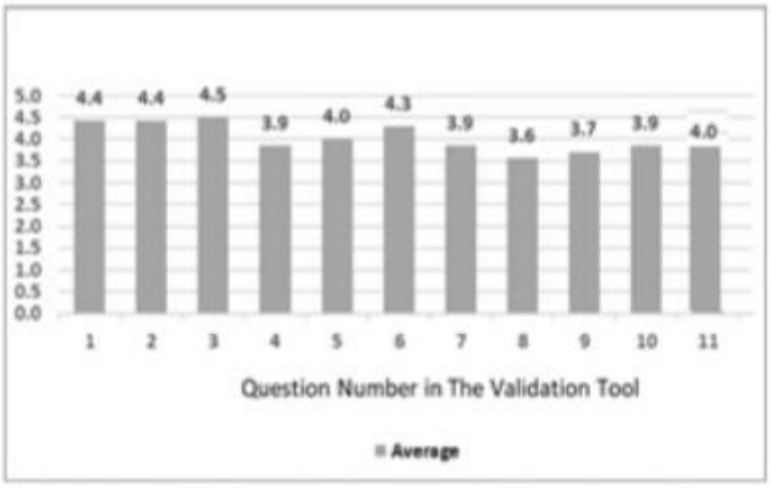

Figure 4. The result of the validation of the proposed taxonomy

Analysis

The "Multi-Criteria Framework for Selecting Software Engineering Tools" offers a structured and objective approach to tool selection, considering various criteria and their relative importance. By incorporating multi-criteria decision-making techniques, it enables organizations to make informed choices, aligning their tool selection with specific needs, and ultimately enhancing the quality and efficiency of software development processes. This framework promotes transparency and strategic alignment in tool selection, addressing the complex and critical decision-making challenges in the software engineering domain

V. RESULTS

When it comes to selecting software testing tools, a multi-criteria framework can be super helpful. It takes into account factors like functionality, usability, cost, support, and more. By considering all these aspects, we can make a well-informed decision and choose the best tools for our needs. It's like having a checklist to guide us in the selection process. This framework helps us save time and effort by narrowing down our options and ensuring we choose tools that align with our requirements. So, it's a smart approach to selecting software testing tools.

VI. CONCLUSION AND FUTURE WORK

In conclusion, a multi-criteria framework for selecting software engineering tools offers a systematic and data-driven approach that enhances decision-making. It enables organizations to align tool choices with project requirements, reducing risks and promoting transparency. However, it may introduce complexity and subjectivity in the process. When applied thoughtfully, this framework empowers organizations to make well-informed tool selections that contribute to project success and long-term efficiency.

For future work, the taxonomy can be simplified. A wide validation for the taxonomy by developers and academics is recommended. An experimental study is recommended as it can provide an accurate image about the proposed taxonomy.

REFERENCES

1. B. B. Agarwal, S. P. Tayal, and M. Gupta, Software Engineering and Testing. Boston, MA, USA: Jones & Bartlett, 2010.
2. S. Bhuvana and M. V. Srinath, ‘‘A survey on automated combinatorial testing for software tool (ACTS) with experimental revise based on T-way test generation,’’ Int. J. Comput. Appl., vol. 6, no. 3, pp. 13–20, 2016.
3. V. Vaishnavi, W. Kuechler, and S. Petter. (Mar. 21, 2017). Design Science Research in Information Systems. [Online]. Available: http://www.desrist.org/design-research-in-information-systems/

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 || A Monthly Peer Reviewed & Referred Journal |

|| Volume 11, Issue 10, October 2023 ||

| DOI: 10.15680/IJIRCCE.2023.1110039 |

4. E. Dustin, J. Rashka, and J. Paul, Automated Software Testing: Introduction, Management, and Performance. Boston, MA, USA: Addison-Wesley, 1999.
5. T. Garrepalli, ‘‘Knowledge Management in software testing,’’ M.S. thesis, Blekinge Inst. Technol., Karlskrona, Sweden, 2015.
6. D. R. Mohammad, S. Al-Momani, Y. M. Tashtoush, and M. Alsmirat, ‘‘A comparative analysis of quality assurance automated testing tools for Windows mobile applications,’’ in Proc. IEEE 9th Annu. Comput. Commun. Workshop Conf. (CCWC), Jan. 2019, pp. 414–419.
7. H. Anjum, M. Imran, M. Jehanzeb, M. Khan, S. Chaudhry, S. Sultana, Z. Shahid, F. Zeshan, and S. Nazir, ‘‘A comparative analysis of quality assurance of mobile applications using automated testing tools,’’ Int. J. Adv. Comput. Sci. Appl., vol. 8, no. 7, pp. 249–255, 2017.

INNO SPACE
SJIF Scientific Journal Impact Factor

Impact Factor: 8.379

doi cross ref

ISSN INTERNATIONAL STANDARD SERIAL NUMBER INDIA

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details