EDITORIAL

Software Engineering in the Era of Intelligence, Security, and Automation

Shabib Aftab[1,*]

[1] Department of Computer Science, Virtual University of Pakistan, Lahore 54000, Pakistan

Abstract

This editorial introduces the *ICCK Journal of Software Engineering* (*JSE*) as an academic platform dedicated to advancing research and innovation across the full spectrum of software engineering. The journal aims to create an inclusive and high-quality space for contributions that span from core theoretical foundations to the latest practical advancements, with a strong focus on emerging technologies. Software engineering as a discipline continues to face a wide range of unresolved challenges despite its critical role in shaping the digital world. Limitations in scalability, adaptability, integration of intelligent systems, and the gap between academic research and real-world application still persist. This editorial is dedicated to highlighting these shortcomings, not only to reflect on them but to advocate for renewed focus and innovative strategies that can move the field forward. Key focus areas include AI-assisted development, continuous integration and deployment, secure and ethical software lifecycles, intelligent code generation using large language models, and the adoption of sustainable, human-centered engineering practices. *JSE* is committed to fostering reproducible, ethical, and impactful research. In outlining the motivation behind the journal, this editorial highlights the urgent challenges facing the field and identifies key thematic domains to be emphasized. We extend a warm invitation to authors, reviewers, and collaborators across academia and industry to engage with this vision. By fostering a rigorous yet collaborative publication environment, the journal aspires to become a vital resource for the broader research and professional community.

Keywords: software engineering, AI-assisted development, secure software lifecycle, ethical software development, intelligent systems.

1 Introduction

In the past two decades, software engineering has evolved from static development practices into a dynamic, intelligent, and automated discipline that touches every sector of modern life. It forms the backbone of modern technology, powering everything from financial systems and healthcare platforms to mobile apps and intelligent devices. Traditional models like Waterfall and object-oriented design paved the way, but today's practices have expanded to include Agile methodologies, AI-assisted development, intelligent testing, low-code/no-code platforms, and

Submitted: 04 July 2025
Accepted: 05 July 2025
Published: 07 July 2025

Vol. 1, **No.** 1, 2025.
10.62762/JSE.2025.534855

***Corresponding author:**
✉Shabib Aftab
shabib.aftab@gmail.com

Citation
Aftab, S. (2025). Software Engineering in the Era of Intelligence, Security, and Automation. *ICCK Journal of Software Engineering*, 1(1), 1–8.

large language models [1]. The integration of machine learning in software engineering, explainable AI (XAI), and secure development practices reflect the field's growing complexity and reach [2].

Despite these advancements, notable challenges continue to persist across the software engineering landscape. Core attributes such as scalability, maintainability, explainability, and ethical development remain difficult to fully achieve, particularly as systems grow in size and complexity [3]. While automation has accelerated various stages of the software lifecycle many existing tools are limited in scope, lack generalizability, or fail to adapt across diverse environments. AI-driven solutions, though promising, often suffer from issues related to reliability, transparency, and trustworthiness, especially when deployed in safety-critical or ethically sensitive domains [4]. Furthermore, the persistent disconnect between academic research and industry adoption hampers the translation of theoretical innovations into practical, scalable solutions [5]. These challenges underscore the urgent need for renewed focus, deeper collaboration, and more robust methodologies that can effectively guide the next phase of progress in software engineering.

As the scope and complexity of software systems expand, so does the need for scholarly venues that support rigorous research and promote real-world applicability [6]. It is with this spirit that we proudly introduce the *ICCK Journal of Software Engineering* (*JSE*), a platform committed to publishing high-quality research spanning the broad and dynamic spectrum of software engineering. This journal aims to address the evolving challenges and opportunities in software engineering by offering an inclusive platform for academics, practitioners, and industry experts alike. As Editor-in-Chief, it is my honor to set the stage for what we hope will become a respected and impactful journal in the global academic and professional community. This inaugural issue marks the beginning of that mission. We warmly welcome researchers, practitioners, and thought leaders from across sectors to join us in addressing today's challenges and shaping tomorrow's software engineering landscape through high-quality, forward-thinking contributions.

2 Scope of the Journal

The *JSE* aims to publish high-quality, peer-reviewed research and review articles that contribute to the advancement of software engineering in both theory and practice. The journal serves as a platform for researchers, practitioners, and educators to share innovative ideas, methodologies, tools, and frameworks that address current and emerging challenges in software development.

2.1 Core Areas of Interest

The journal welcomes contributions from a wide range of software engineering subfields, including but not limited to:

- Software Development Life Cycle (SDLC)
- Programming Paradigms and Tools
- Software Quality and Reliability
- AI in Software Engineering
- Human Aspects and Management
- Secure and Trustworthy Software Systems
- Interdisciplinary and Industry-Oriented Software Engineering
- Large-Scale, Distributed, and Sustainable Software Systems
- Emerging and Cross-Disciplinary Topics

The *JSE* welcomes diverse and original work that pushes the boundaries of how software is designed, built, and maintained. The journal aims to bring together researchers and practitioners who are passionate about solving real-world problems through innovative and thoughtful software engineering approaches.

3 Trends and Challenges in Software Engineering

Software engineering is growing and changing quickly due to new technologies, rising user expectations, and the increasing complexity of modern systems. As industries continue to go digital, developers are expected to build faster, more secure and user-friendly applications. At the same time, developers face ongoing challenges such as managing outdated code, handling security threats, and adapting to remote work environments. These rapid changes and persistent challenges highlight the need for smarter tools and continuous learning among professionals, researchers, and educators in the field.

3.1 Emerging Trends

To meet the growing demands of modern software systems, engineers are adopting innovative tools and frameworks that offer practical solutions to long-standing problems [7–10]. From AI-powered development assistants to environmentally conscious coding practices, these advances are reshaping how software is designed, built, and deployed in real-world environments [11]. The following highlights some of the most significant trends shaping the discipline today.

- *AI-Driven Development*: Artificial Intelligence is changing software development by automating coding tasks. Tools like GitHub Copilot help developers write and debug code faster [12]. AI can translate old systems into new languages efficiently. These technologies improve productivity and reduce human error. Developers can focus more on creative problem-solving. AI is becoming essential in modern software workflows.

- *Low-Code/No-Code Platforms*: Low-code and no-code platforms let non-developers create software easily. They use drag-and-drop interfaces to simplify application building. This speeds up development and reduces costs. Businesses adopt these platforms to quickly launch new solutions [13]. They also encourage collaboration between technical and business teams. These platforms are rapidly growing in popularity worldwide.

- *Sustainable Software Practices*: Sustainable software aims to reduce environmental impact. Developers write energy-efficient code to save power. Techniques like carbon-aware load balancing are becoming common [14]. Optimizing build processes helps lower energy use. These efforts support global sustainability goals. Green coding is now a key focus in software development.

- *Cybersecurity Integration*: Security is integrated throughout software development to combat rising cyber threats. DevSecOps embeds security checks early in the process. Zero Trust architecture assumes no device is automatically trusted. This limits access and prevents breaches. Automated tools help detect vulnerabilities continuously [15]. Strong security practices ensure safer software products.

- *Edge Computing*: Edge computing processes data near its source to reduce delay. It supports real-time applications like healthcare monitoring. Smart cities use edge devices to improve public services. Software must be optimized for limited resources in these environments [16]. This approach reduces network congestion. Edge computing is transforming how applications run and interact.

3.2 Current Challenges in Software Engineering

The software engineering landscape is evolving rapidly, but several critical challenges persist. These challenges impact software quality, security, team dynamics, and cost management [17]. Addressing them is vital for sustainable growth and innovation in the field [18]. Below are some key issues currently faced by the industry. Understanding these problems helps shape better practices and research. The following points highlight the major challenges in software engineering today.

- *AI-Generated Code Vulnerabilities*: While AI tools accelerate development, they can introduce security vulnerabilities. Ensuring the safety and reliability of AI-generated code remains a significant challenge [19].

- *Managing Technical Debt*: Rapid development cycles and evolving requirements often lead to accumulated technical debt. Managing and refactoring this debt is crucial to maintain software quality over time.

- *Software Supply Chain Security*: High-profile attacks (e.g., SolarWinds, Log4Shell) have shown that vulnerabilities in third-party libraries and dependencies can compromise entire systems. Securing the software supply chain has become a top priority [20].

- *Cloud Cost Management*: Organizations are facing budget overruns in cloud expenditures. Effective forecasting, monitoring, and optimization strategies are essential to control costs.

- *Remote and Hybrid Work Complexities*: The shift to remote/hybrid models introduced communication issues, productivity tracking challenges, and gaps in team cohesion. Building strong engineering cultures and effective collaboration tools is now essential.

3.3 From Challenges to Solutions

To overcome today's challenges and build a resilient future for software engineering, the following key directions are essential

- Encourage the use of emerging technologies while ensuring responsible and ethical practices.
- Focus on building systems that are secure, scalable, and adaptable to change.
- Use intelligent automation to boost productivity and reduce manual errors.
- Improve collaboration and communication in hybrid and remote work settings.
- Prioritize software quality through proper testing, refactoring, and lifecycle management.

4 Vision and Mission of the Journal

The *JSE* is founded on a commitment to advancing the evolving discipline of software engineering through high-impact, ethically sound, and forward-looking research. This section outlines the journal's long-term vision and operational mission that guide its publishing philosophy. The *JSE* envisions becoming a globally recognized platform that:

- Leads the advancement of software engineering by showcasing research that is technically innovative, socially relevant, and ethically responsible.
- Bridges the longstanding gap between academic research and real-world application by encouraging scalable and reproducible work.
- Cultivates a dynamic community where emerging technologies such as artificial intelligence, explainable systems, and intelligent automation are explored within the context of modern software engineering.
- Supports the development of secure, sustainable, and intelligent software systems that address diverse societal and industrial needs.

To achieve its vision, The *JSE* is dedicated to the following:

- Publishing high-quality, peer-reviewed research in both foundational and emerging areas of software engineering.
- Fostering interdisciplinary dialogue that integrates fields such as artificial intelligence, systems design, HCI, and data science.
- Encouraging real-world relevance by supporting applied research that addresses concrete problems faced by industry and society.
- Providing a platform for collaboration among researchers, practitioners, educators, and developers across academia and industry.

4.1 Core Principles and Ethics

The *JSE* is firmly dedicated to maintaining the highest levels of honesty, transparency, and scholarly excellence throughout the publication process. Upholding strong ethical standards is essential to ensure that all published research meaningfully advances the software engineering discipline [21].

- *Integrity and Scientific Rigor*: Submissions undergo a comprehensive and fair peer review to verify that research is original, valid, and methodologically sound. Authors are expected to conduct and present their work with honesty and precision, following recognized best practices [22].
- *Commitment to Diversity and Inclusion*: Recognizing the value of varied perspectives, the journal actively welcomes contributions from researchers across different geographic regions, career stages, and cultural backgrounds. Promoting an inclusive scholarly environment ensures that a broad range of ideas and innovations enrich the field [23].
- *Innovation and Knowledge Advancement*: The journal is committed to promoting original thinking and forward-looking research that addresses real-world challenges in software engineering. It encourages the exploration of emerging technologies, novel methodologies, and interdisciplinary approaches.
- *Fostering Collaboration and Professionalism*: The journal champions respectful interaction and constructive feedback among authors, reviewers, and editors. By encouraging open, collegial communication and cooperation, the *JSE* supports a collaborative atmosphere that drives progress and excellence in software engineering research [24].

4.2 Strengthening Academic Participation

To help build a strong and inclusive research community in software engineering, the *JSE* is committed to several key initiatives. The journal will publish special issues on emerging and high-impact topics such as AI in software, blockchain applications and cloud computing. It will recognize outstanding contributions through best paper awards and will also highlight the achievements of early-career researchers to encourage and support new talent. By collaborating with conferences, workshops, and academic organizations, the journal aims to create valuable networking and mentoring opportunities for researchers at all levels. Additionally, it actively promotes the involvement of underrepresented groups to ensure diverse perspectives and equal participation across the academic community.

4.3 Envisioning the Future of Software Engineering

The future of software engineering lies in creating systems that are not only intelligent and efficient but also ethical and sustainable [25]. As technologies like AI and quantum computing continue to evolve, software engineers will play a crucial role in shaping how these innovations are built, deployed, and maintained. There will be a growing need for adaptive methodologies and closer collaboration between academia, industry, and policy-makers. By embracing continuous learning, responsible innovation, and global collaboration, the field is poised to address complex challenges and deliver impactful solutions.

5 Editorial Strategy

The *JSE* is committed to advancing the field through rigorous, transparent, and ethical publishing practices. Our editorial approach ensures that every manuscript is carefully reviewed, helping us share high-quality research that makes a valuable contribution to the software engineering field.

5.1 Manuscript Handling and Review Process

All submitted manuscripts follow a structured workflow to ensure fairness, quality, and academic integrity. From initial screening to peer review and final decision, each stage is carefully managed by the editorial team.

5.1.1 Article Submission

Authors submit their manuscripts through the journal's online submission system, following the specified formatting and ethical guidelines. All required elements—such as title page, abstract, figures, and references—must be properly included. The system confirms receipt and assigns a unique tracking ID to the submission. Submissions are then forwarded to the Managing Editor for the initial evaluation.

5.1.2 Initial Screening

The Managing Editor reviews the manuscript to ensure it fits the journal's scope and meets technical standards. Basic checks are performed for formatting, completeness, originality, and compliance with ethical policies. Manuscripts that fail to meet these requirements are either desk-rejected or returned to the authors for revision. Only submissions that pass this screening move forward to peer review.

5.1.3 Peer Review Process

Once approved, the manuscript is assigned to a Handling Editor who oversees the peer review process. The Handling Editor selects at least two qualified reviewers and ensures no conflicts of interest exist. A single-blind review model is followed—reviewers know the authors' identity, but not vice versa. Reviewers evaluate the manuscript for originality, soundness, novelty and ethical standards. Their feedback is collected and used to inform editorial decisions.

5.1.4 Review Outcome and Editorial Decision

Based on the reviewers' reports, the Handling Editor may recommend acceptance, minor/major revision, or rejection. Authors are informed of the decision along with reviewer comments and are invited to revise the manuscript accordingly. Revised manuscripts may undergo additional review if needed. The Handling Editor ensures that the revisions address all major concerns. The final recommendation is submitted to the Editor-in-Chief for the final decision.

5.1.5 Final Acceptance and Publication

Once a manuscript is accepted, it goes through copyediting, formatting, and final proofreading. Authors review the final proofs to ensure accuracy before publication. The article is then published online with a DOI and later assigned to a regular journal issue. The journal ensures timely and high-quality publication of all accepted articles. This process guarantees that the research reaches the academic community in a polished and accessible format.

5.2 Research Ethics and Publication Integrity

The *JSE* follows the ethical principles outlined by the International Committee of Medical Journal Editors (ICMJE) and aligns its editorial practices with the Code of Conduct and Best Practice Guidelines recommended by the Committee on Publication Ethics (COPE):

- *Authorship and Contribution Ethics*: Only those who significantly contribute to the research should be credited as authors. Each author must approve the final version before submission.

- *Plagiarism and Originality*: All submitted work must be original and properly cited. Copying text, figures, or ideas without attribution is considered plagiarism.

- *Data Transparency and Reproducibility*: Authors must clearly describe their data collection and analysis methods. Raw data should be stored securely and shared if requested for verification.

- *Duplicate Submissions and Falsification*: Manuscripts must not be submitted to multiple journals at the same time. Altering data, results, or images is considered serious misconduct. Such actions lead to rejection or retraction of the article.

5.3 Acknowledging Peer Review and Editorial Excellence

The *JSE* values the essential role of peer reviewers and editorial board members in upholding the journal's academic quality. To recognize their efforts, exceptional reviewers are acknowledged annually for their contributions. Reviewer activity can also be credited via platforms such as Publons or ORCID upon request. The journal also accepts nominations for new editorial board members and guest editors, particularly for special issues aligned with its scope. Editorial Board members contribute by setting the journal's scholarly direction, ensuring rigorous peer review, and maintaining ethical and publication standards. Their commitment ensures the journal remains credible, inclusive, and aligned with current research trends.

5.4 Openness and Collaboration

The *JSE* offers an open access option to make research freely available to a global audience. This openness promotes wider dissemination and greater visibility of published work, accelerating knowledge sharing and impact. Transparency in sharing data, code, and methodologies supports reproducibility and builds trust in scientific findings. Collaboration among authors, reviewers, and editors strengthens the research community and fosters innovation. An inclusive environment encourages diverse perspectives and interdisciplinary partnerships that enrich software engineering research. Open dialogue and constructive feedback enhance the quality and rigor of published work. By embracing openness, the *JSE* cultivates a culture of shared learning and continuous advancement.

6 Call for Contributions and Future Directions

The *JSE* offers an open access option to make research freely available to a global audience. This openness promotes wider dissemination and greater visibility of published work, accelerating knowledge sharing and impact. Transparency in sharing data, code, and methodologies supports reproducibility and builds trust in scientific findings. Collaboration among authors, reviewers, and editors strengthens the research community and fosters innovation. An inclusive environment encourages diverse perspectives and interdisciplinary partnerships that enrich software engineering research. Open dialogue and constructive feedback enhance the quality and rigor of published work. By embracing openness, the *JSE* cultivates a culture of shared learning and continuous advancement.

6.1 Invitation to Authors, Reviewers, and Contributors

As the journal enters its formative years, we warmly invite contributions from the global research community. Submissions from senior academics, early-career researchers, and software industry professionals are all welcomed. Interdisciplinary research that pushes boundaries and explores innovative directions in software engineering is especially valued. Qualified experts are invited to join the pool of reviewers and associate editors, as their expertise plays a crucial role in upholding scholarly standards and fostering a culture of constructive feedback. Proposals for special issues on contemporary topics or collaborative workshops in emerging areas of software engineering are also sought. Such collective efforts will help ensure the journal remains current, dynamic, and relevant.

6.2 Global Engagement and Collaboration Opportunities

The *JSE* supports a strong research community by offering a range of academic and professional initiatives that promote global collaboration, recognize emerging talent, and advance the field of software engineering:

- *Guest Editorship Opportunities* for leading experts and early-career researchers.

- *Reviewer and Editorial Board Memberships* to promote active scholarly involvement.

- *Cross-disciplinary Research Collaborations* that bridge academia and industry.

6.3 Thematic Collections and Special Issues

The Journal of Software Engineering actively supports the publication of special issues that explore emerging trends, critical challenges, or alternative perspectives in the field. These collections may focus on cutting-edge topics such as AI-assisted software development, secure DevOps practices, green and sustainable software engineering, or software engineering for cloud-native and edge systems. Guest-edited special issues provide a platform for in-depth exploration of evolving research areas and foster collaboration across academia and industry.

7 Conclusion

The launch of the *ICCK Journal of Software Engineering* is an exciting step toward creating a space that truly celebrates the diversity and creativity within software engineering research. This journal is more than just a publication—it's a community that encourages fresh ideas, collaboration, and research that makes a real difference in society. As software continues to reshape everything around us—from how we learn and heal to how we work and play—this journal hopes to be part of that positive change.

By welcoming researchers, practitioners, and industry experts from all walks of life and corners of the world, the journal offers a place where new perspectives come together and innovative solutions emerge. The goal is to support work that pushes boundaries, sparks progress, and helps build a better future through technology.

We look forward to your ideas, your discoveries, and your partnership in growing this vibrant community. Together, we can make the journal a trusted home for high-quality, impactful software engineering research that speaks to today's challenges and tomorrow's possibilities. We warmly invite you to join us on this exciting journey to shape the future of software engineering together.

Data Availability Statement

Not applicable.

Funding

This work was supported without any funding.

Conflicts of Interest

The author declares no conflicts of interest.

Ethical Approval and Consent to Participate

Not applicable.

References

[1] Zheng, Z., Ning, K., Zhong, Q., Chen, J., Chen, W., Guo, L., ... & Wang, Y. (2025). Towards an understanding of large language models in software engineering tasks. *Empirical Software Engineering, 30*(2), 50. [Crossref]

[2] Hou, X., Zhao, Y., Liu, Y., Yang, Z., Wang, K., Li, L., ... & Wang, H. (2024). Large language models for software engineering: A systematic literature review. *ACM Transactions on Software Engineering and Methodology, 33*(8), 1-79. [Crossref]

[3] Fan, A., Gokkaya, B., Harman, M., Lyubarskiy, M., Sengupta, S., Yoo, S., & Zhang, J. M. (2023, May). Large language models for software engineering: Survey and open problems. In *2023 IEEE/ACM International Conference on Software Engineering: Future of Software Engineering (ICSE-FoSE)* (pp. 31-53). IEEE. [Crossref]

[4] Fan, A., Gokkaya, B., Harman, M., Lyubarskiy, M., Sengupta, S., Yoo, S., & Zhang, J. M. (2023, May). Large language models for software engineering: Survey and open problems. In *2023 IEEE/ACM International Conference on Software Engineering: Future of Software Engineering (ICSE-FoSE)* (pp. 31-53). IEEE. [Crossref]

[5] Yang, J., Jimenez, C. E., Wettig, A., Lieret, K., Yao, S., Narasimhan, K., & Press, O. (2024). Swe-agent: Agent-computer interfaces enable automated software engineering. *Advances in Neural Information Processing Systems, 37*, 50528-50652.

[6] Martínez-Fernández, S., Bogner, J., Franch, X., Oriol, M., Siebert, J., Trendowicz, A., ... & Wagner, S. (2022). Software engineering for AI-based systems: a survey. *ACM Transactions on Software Engineering and Methodology (TOSEM), 31*(2), 1-59. [Crossref]

[7] Ahmed, T., & Devanbu, P. (2022, May). Multilingual training for software engineering. In *Proceedings of the 44th International Conference on Software Engineering* (pp. 1443-1455). [Crossref]

[8] Mandal, A. K., Nadim, M., Roy, C. K., Roy, B., & Schneider, K. A. (2025). Quantum software engineering and potential of quantum computing in software engineering research: a review. *Automated Software Engineering, 32*(1), 27. [Crossref]

[9] Lu, Q., Zhu, L., Xu, X., Whittle, J., & Xing, Z. (2022, May). Towards a roadmap on software engineering for responsible AI. In *Proceedings of the 1st International Conference on AI Engineering: software Engineering for AI* (pp. 101-112). [Crossref]

[10] Lenberg, P., Feldt, R., Gren, L., Wallgren Tengberg, L. G., Tidefors, I., & Graziotin, D. (2024). Qualitative

software engineering research: Reflections and guidelines. *Journal of Software: Evolution and Process, 36*(6), e2607. [Crossref]

[11] Verdecchia, R., Engström, E., Lago, P., Runeson, P., & Song, Q. (2023). Threats to validity in software engineering research: A critical reflection. *Information and Software Technology, 164*, 107329. [Crossref]

[12] Melnikov, A., Kordzanganeh, M., Alodjants, A., & Lee, R. K. (2023). Quantum machine learning: from physics to software engineering. *Advances in Physics: X, 8*(1), 2165452. [Crossref]

[13] Xia, C. S., Deng, Y., Dunn, S., & Zhang, L. (2024). Agentless: Demystifying llm-based software engineering agents. *arXiv preprint arXiv:2407.01489*. [Crossref]

[14] Zhang, K., Yao, W., Liu, Z., Feng, Y., Liu, Z., Murthy, R., ... & Xiong, C. (2024). Diversity empowers intelligence: Integrating expertise of software engineering agents. *arXiv preprint arXiv:2408.07060*.

[15] Von der Mosel, J., Trautsch, A., & Herbold, S. (2022). On the validity of pre-trained transformers for natural language processing in the software engineering domain. *IEEE Transactions on Software Engineering, 49*(4), 1487-1507. [Crossref]

[16] Russo, D., & Stol, K. J. (2021). PLS-SEM for software engineering research: An introduction and survey. *ACM Computing Surveys (CSUR), 54*(4), 1-38. [Crossref]

[17] Lu, Q., Zhu, L., Xu, X., Whittle, J., Douglas, D., & Sanderson, C. (2022, May). Software engineering for responsible AI: An empirical study and operationalised patterns. In *Proceedings of the 44th International Conference on Software Engineering: Software Engineering in Practice* (pp. 241-242). [Crossref]

[18] Zhang, Q., Zhang, T., Zhai, J., Fang, C., Yu, B., Sun, W., & Chen, Z. (2023). A critical review of large language model on software engineering: An example from chatgpt and automated program repair. *arXiv preprint arXiv:2310.08879*.

[19] Wohlin, C. (2021). Case Study Research in Software Engineering—It is a Case, and it is a Study, but is it a Case Study?. *Information and Software Technology, 133*, 106514. [Crossref]

[20] Giray, G. (2021). A software engineering perspective on engineering machine learning systems: State of the art and challenges. *Journal of Systems and Software, 180*, 111031. [Crossref]

[21] Shin, J., Tang, C., Mohati, T., Nayebi, M., Wang, S., & Hemmati, H. (2023). Prompt engineering or fine tuning: An empirical assessment of large language models in automated software engineering tasks. *arXiv preprint arXiv:2310.10508*.

[22] Moreira, A., Lago, P., Heldal, R., Betz, S., Brooks, I., Capilla, R., ... & Venters, C. C. (2025). A roadmap for integrating sustainability into software engineering education. *ACM Transactions on Software Engineering and Methodology, 34*(5), 1-27. [Crossref]

[23] Robillard, M. P., Arya, D. M., Ernst, N. A., Guo, J. L., Lamothe, M., Nassif, M., ... & Stol, K. J. (2024). Communicating study design trade-offs in software engineering. *ACM Transactions on Software Engineering and Methodology, 33*(5), 1-10. [Crossref]

[24] Akdur, D. (2022). Analysis of software engineering skills gap in the industry. *ACM Transactions on Computing Education, 23*(1), 1-28. [Crossref]

[25] Ahmad, M. O., & Gustavsson, T. (2024). The Pandora's box of social, process, and people debts in software engineering. *Journal of Software: Evolution and Process, 36*(2), e2516. [Crossref]

Shabib Aftab received his Ph.D. in Computer Science from the National College of Business Administration and Economics, Lahore. He holds an M.S. degree in Computer Science from Comsats University, Islamabad, an M.Sc. in Information Technology from Punjab University College of Information Technology, Lahore, and a Bachelor's degree from Government College University, Lahore. With over 16 years of academic and research experience, he is currently serving as an Assistant Professor in the Department of Computer Science at the Virtual University of Pakistan.

Dr. Aftab's research interests span Software Process Improvement, Data Mining, Predictive Analytics, Applied Machine Learning, and Applied Data Science. His work emphasizes developing innovative, data-driven solutions to real-world challenges in domains including healthcare, finance, and software engineering. He has authored over 60 research publications in renowned international journals and conferences and is a Senior Member of IEEE, reflecting his active engagement in advancing scientific research and contributing to the global academic community. (Email: shabib.aftab@gmail.com)