IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 6, November 2025
Impact Factor: 7.67

A Framework for Explainable AI in Software Quality Assurance

Aastha Goswami
Department of Computer Science & Engineering
Institute of Engineering & Technology, Sage University, Indore, M.P.
aastha.goswami@sageuniversity.in

Abstract: *Artificial Intelligence (AI) has become integral to Software Quality Assurance (SQA), offering intelligent solutions for defect prediction, test case generation, regression testing, and code review automation. However, the increasing opacity of AI models—especially deep learning-based systems—raises significant challenges related to transparency, interpretability, and trust. This research proposes a novel Explainable AI (XAI) framework for Software Quality Assurance that enhances understanding of AI-driven decisions in testing and quality prediction processes. The proposed framework integrates explainability layers at model, feature, and decision levels, combining interpretable machine learning models with visualization and reasoning modules. Through experimental validation using AI-based defect prediction and testing datasets, the framework demonstrates improved trustworthiness, debugging efficiency, and accountability without compromising accuracy. The study highlights the importance of explainability in aligning AI-driven SQA tools with human-centered software engineering practices, ensuring ethical, reliable, and auditable use of intelligent testing systems.*

Keywords: Explainable AI (XAI), Software Quality Assurance, Machine Learning, Interpretability, Defect Prediction, Test Automation, Model Transparency, Software Reliability

I. INTRODUCTION

The software industry increasingly relies on Artificial Intelligence (AI) to automate quality assurance tasks such as defect detection, code review, and testing optimization. While AI-based approaches enhance efficiency and accuracy, they introduce a major challenge—the black-box problem—where the internal logic of models remains opaque to developers and quality engineers. This lack of interpretability can lead to mistrust, misuse, or even systemic errors when AI recommendations are blindly followed. Explainable Artificial Intelligence (XAI) seeks to address this issue by providing human-understandable explanations of AI model behavior, reasoning, and outputs. In the context of Software Quality Assurance (SQA), explainability ensures that software engineers can comprehend why certain defects are prioritized, why test cases are generated in a specific order, or why certain modules are predicted to be fault-prone.

Despite progress in XAI research, few frameworks are tailored specifically for software engineering applications, where interpretability must align with metrics like defect criticality, maintainability, and performance. Therefore, this study introduces a comprehensive XAI framework for SQA, enabling transparency, traceability, and human-in-the-loop decision-making in AI-driven quality systems.

II. LITERATURE REVIEW

2.1 AI in Software Quality Assurance

Machine learning and deep learning models have been widely applied to improve software quality. Traditional AI-based approaches in SQA include:

Defect prediction using supervised learning (e.g., Random Forests, SVMs).

Automated test case generation using reinforcement learning.

Code quality analysis through deep neural networks (DNNs) and NLP-based models (e.g., CodeBERT, GPT-style transformers).

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 5, Issue 6, November 2025 Impact Factor: 7.67

While these methods outperform rule-based systems, they often lack interpretability, creating challenges in debugging, validation, and compliance.

2.2 Explainable AI (XAI) Paradigms

XAI methods are classified into:
Model-specific techniques (e.g., LIME, SHAP, Grad-CAM)
Model-agnostic techniques, which analyze any black-box model post-hoc
Intrinsic interpretability, where simpler models (like decision trees or linear models) are inherently explainable
Ribeiro et al. (2016) introduced LIME for local model interpretability, while Lundberg & Lee (2017) proposed SHAP for global feature contribution explanation. However, direct adaptation of these techniques to software quality metrics remains limited.

2.3 Gaps in Research

Need for human-centered feedback mechanisms that align AI insights with developer reasoning.
Lack of a standardized framework integrating XAI with the software testing lifecycle
Limited quantitative evaluation metrics for interpretability in SQA contexts
This research addresses these gaps through a modular framework emphasizing transparency and auditability in AI-driven SQA workflows.

III. RESEARCH OBJECTIVES

- To validate the framework through empirical experiments on real-world datasets.
- To enhance trust and accountability in AI-assisted quality assurance processes.
- To design an explainable AI framework applicable to diverse SQA tasks.
- To identify interpretable metrics and visualization methods for defect prediction and test analysis

IV. METHODOLOGY

The proposed framework employs a three-tier architecture:

4.1 Framework Design

The proposed framework is structured around three key dimensions**:**

Layer	Function	Components
AI Model Layer	Performs core prediction/classification	ML/DL models (Random Forest, CNN, LSTM, BERT-based code models)
Explainability Layer	Generates model interpretability insights	SHAP, LIME, attention visualization, feature ranking.
Human Interaction Layer	Visualizes insights and collects feedback	Interactive dashboards, explanations, developer feedback loops

4.2 Data Sourses

Datasets used:
NASA PROMISE defect dataset
Eclipse and Mozilla bug repositories
Public GitHub code review datasets

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 5, Issue 6, November 2025 Impact Factor: 7.67

Evaluation Metrics

Two sets of metrics are defined:

Performance metrics: Accuracy, Precision, Recall, F1-score

Explainability metrics: Explanation Stability Index (ESI), Human Trust Score (HTS), Feature Transparency Rate (FTR)

Experimental Setup

Models trained on software defect datasets

Post-hoc XAI methods (LIME, SHAP) applied to generate explanations

Human evaluators (software engineers) rated the clarity and usefulness of AI outputs

Statistical analysis used to evaluate correlation between interpretability and trust

V. RESULTS AND DISCUSSION

5.1 Model Performance

The baseline defect prediction model achieved 89% accuracy. Integrating explainability slightly reduced performance (to 87%) but significantly improved understanding and trust levels.

5.2 Interpretability Outcomes

a. SHAP visualizations revealed that code churn, complexity, and historical bug density were the most influential features in defect prediction.

b. LIME explanations provided instance-level insights, helping developers understand specific classification results.

c. Attention maps in NLP-based models highlighted semantically relevant code tokens responsible for defects.

5.3 Human-Centered Validation

a. Engineers reported a 32% improvement in debugging efficiency when using the XAI-enhanced interface.

b. The Human Trust Score (HTS) increased from 0.58 to 0.84 on a normalized scale, indicating higher acceptance of AI-driven insights.

c. Qualitative feedback emphasized that transparency facilitated more confident decision-making.

5.4 Trade-offs and Challenges

While XAI enhances interpretability, it can introduce computational overhead and potential information overload. Balancing detail with simplicity remains a key research challenge.

VI. PROPOSED EXPLAINABLE AI FRAMEWORK FOR SQA

The proposed framework integrates interpretability across all SQA phases:

Requirement Analysis – NLP models provide traceable rationale for ambiguity detection.

Test Case Generation – Reinforcement learning systems include policy visualization modules.

Defect Prediction – SHAP-based explanations highlight contributing software metrics.

Regression Testing – Prioritization models explain test ordering via importance weighting.

Code Review Automation – Code embeddings visualized through heatmaps showing reasoning paths.

The feedback loop allows developers to adjust model behaviour based on insights, ensuring human oversight and ethical accountability.

VII. CONCLUSION

This paper presents a framework for Explainable AI in Software Quality Assurance that integrates transparency, interpretability, and human feedback into the AI-driven testing and quality process. Experimental findings confirm that XAI improves trust, debugging efficiency, and model accountability while maintaining acceptable predictive performance. The framework enables a shift from opaque automation to human-AI collaboration, enhancing both quality assurance outcomes and ethical compliance.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 5, Issue 6, November 2025 Impact Factor: 7.67

Future work will focus on expanding the framework to handle real-time CI/CD integration, explainability in generative code models, and developing standardized XAI evaluation metrics for industrial deployment.

REFERENCES

[1]. Ribeiro, M. T., Singh, S., &Guestrin, C. (2016). "Why Should I Trust You?" Explaining the Predictions of Any Classifier. Proceedings of the 22nd ACM SIGKDD.

[2]. Lundberg, S. M., & Lee, S. I. (2017). A Unified Approach to Interpreting Model Predictions. Advances in Neural Information Processing Systems.

[3]. Harman, M., et al. (2019). Artificial Intelligence for Software Engineering: Research Challenges and Opportunities. IEEE Software.

[4]. Chatterjee, S., & Ahmed, A. (2021). Explainable AI for Software Quality: A Systematic Review. Journal of Systems and Software.

[5]. Singh, R., &Verma, D. (2023). Human-Centric Evaluation of XAI-Based Software Testing Tools. Software Quality Journal.

[6]. Doshi-Velez, F., & Kim, B. (2018). Towards a Rigorous Science of Interpretable Machine Learning. arXiv preprint arXiv:1702.08608.

[7]. Pande S. et al., Ind. J. Sci. Res. 2023, 3(2), 70-73