

EXPLAINABLE AND ETHICAL ARTIFICIAL INTELLIGENCE IN SOFTWARE ENGINEERING: A REVIEW OF RECENT ADVANCES

Amna Bibi^{*1}, Hasna Arshad², Abdullah Shahroze³, Muhammad Hamza⁴,
Muhammad Ubaid Ur Rehman⁵

^{*1,4}Student HITEC University Taxila

²Assistant professor HITEC University Taxila

^{3,5}Lecturer HITEC University Taxila

^{*1}amnaniazi003@gmail.com, ²hasna.arshad@hitecuni.edu.pk, ³abdullah.shahrose@hitecuni.edu.pk,
⁴hamza67661@gmail.com, ⁵mubaid63@gmail.com

DOI: <https://doi.org/10.5281/zenodo.22829316>

Keywords

Explainable Artificial Intelligence;
Ethical AI; Software Engineering;
Trustworthy AI; Software
Development Lifecycle; Human-
Centered AI

Article History

Received: 01 June 2026

Accepted: 23 August 2026

Published: 18 September 2026

Copyright @Author

Corresponding Author: *

Amna Bibi

Abstract

As AI continues its integration into software development, it has made software development more automatic, intelligent and data-driven throughout the Software Development Life Cycle (SDLC). With the increasing use of AI-based methods, however, there are significant concerns of transparency, trust, fairness, and accountability. Explainable Artificial Intelligence (XAI) and Ethical AI have thus become vital pillars to build on for trustworthy AI software engineering. In this review paper, we provide a structured synthesis of literature on explainable and ethical Artificial Intelligence in Software Engineering (AISE) from 2020 to 2025. AI in software engineering literature is explored from two main perspectives: (i) Explainable AI in Software Engineering and (ii) ethical frameworks for AI and how they are applied in software engineering. It highlights the key areas of dominance such as defect prediction, testing, maintenance, etc., and key limitations like the lack of consistent evaluation of explanations, lack of coverage of early SDLC phases, and gaps between ethical principles and their operational implementation. The results underscore the important relationship between explainability and ethical AI and suggest developer-centric, integrated, lifecycle-aware approaches. This review discusses the trends, challenges, and gaps in the current research and proposes future directions for trustworthy and responsible AI in software engineering.

INTRODUCTION

Modern software systems have evolved to be large, complex, and data intensive, thus raising the challenges of existing rule-based and manual software engineering approaches. Many traditional technologies have trouble keeping large

code bases, changing requirements, and continuous integration systems in check. These challenges have spurred the use of Artificial Intelligence (AI) in software engineering, aiming at intelligent automation, data-informed decision-

making, and adaptable learning throughout the software development lifecycle (SDLC). AI techniques are capable of analyzing a large number of historical software artifacts and identifying the underlying patterns in the software data [19]. These features aid in various aspects of software development, such as defect prediction [2], automated testing, code generation [32] and long-term software maintenance [28], enhancing the overall quality and efficiency of software development.

The digitization of AI in software engineering instruments has revolutionized conventional software development processes, making them more automated, precise, and scalable. Today, AI-based systems are widely used for code recommendation and software code generation [32], vulnerability detection and bug detection [4], automated software debugging and bug fixing [13] and software evolution tasks. Consequently, AI has been integrated into software development in a major way, making it a key aspect of today's software engineering practices. But, numerous AI models are considered black boxes as they make predictions or recommendations without explaining the logic behind the prediction. But in software engineering applications, where the comprehension of the system's behavior and validation of its results are crucial to software reliability and developer confidence, this transparency is a major concern. Furthermore, the lack of explainability may decrease developers' trust of AI-assisted tools and make them less effective in using them [16].

To overcome these issues of transparency, Explainable AI (XAI) has emerged as a key area of study that aims to interpret and understand AI driven decisions to software engineers. For software engineering tools, XAI can expose the behavior of the model, and justify the automatic recommendations [28]. Important software features of defect prediction models can be identified using explanation techniques like SHAP [2] and local explanations of model results using LIME [27]. Rule-based explanations also help developers evaluate the reliability and

correctness of AI-generated insights, along with example-based explanations. The benefits include better explainability which can improve debugging, minimize the dependency on automation without the human touch, and foster a deeper partnership between humans and AI-enabled software. In addition to explainability, there is another dimension of ethical AI that has emerged in AI-powered software engineering: that of ethics. Ethical AI highlights aspects of fairness, accountability, transparency, privacy protection, and risk mitigation throughout the SDLC [23]. AI-powered software tools, however, can either reinforce bias or an unfair decision-making process if it lacks the proper ethical safeguards in its design and use [1]. Empirical research using examples has shown that dealing with fairness-related issues is crucial to prevent the adverse effects of software analytics and decision-making systems [9]. In the context of AI tools, such as those focusing on code reviews, vulnerability detection, and maintenance automation, ethical concerns are particularly significant, with potential impacts on software quality and developer success.

Although explainable and ethical AI has gained a significant amount of focus in the study of AI, there are a few drawbacks. While there are many existing XAI approaches that focus on the technical correctness of the explanation, few have been developed to be usable in a practical setting, making the explanations less useful for developers in real-world settings [15]. Likewise, empirical research has shown that the explanations produced by the different popular XAI techniques do not necessarily align with developers' expectations and processes [27]. Meanwhile, ethical guidelines for AI are generic and rarely are followed by the implementation of tools or into development pipelines [24]. The broader ethical concerns surrounding AI systems, including potential misuse and unintended harm, add another layer of complexity to its application in software engineering scenarios [33]. Furthermore, there is still no unified metrics for evaluation of explanation quality, ethical compliance, and

developer-centered usability that can be used in practical use of explainable and ethical AI solutions [5, 17].

A number of recent studies published in the last five years (2020-2025) emphasize the need to consider explainability and ethical aspects throughout the SDLC spectrum, from requirements engineering, design, implementation, testing, and deployment to maintenance. Systematic literature reviews also underline the importance of embedding XAI and ethical aspects throughout the software engineering processes, making them a core part of the process, not an after-thought [7, 21]. An in-depth analysis of explainable and ethical AI in software engineering is thus necessary to recognize the current trends, unsolved problems and future research direction. Improving transparency, accountability and fairness in AI-powered software tools is essential to progress towards responsible, human-centred and sustainable software engineering practices.

Research Questions (RQs)

- **RQ1 – Explainability Dimension:** To what extent are current XAI techniques effective in understanding, validating, and trusting of AI-driven software engineering tools' recommendations for software engineers?
- **RQ2 – Ethical AI Dimension:** How feasible are it to apply the existing ethical AI principles (fairness, accountability, transparency) in AI software engineering tools and what are the obstacles in that regard?
- **RQ3 – Integration & Gaps Across SDLC:** What are the gaps in SDLC in terms of sufficient explainable and ethical support of AI in various phases of SDLC?

2. Literature Review

In this section, we review and synthesize the most relevant and influential works on explainable and ethical AI in software engineering tools from 2020–2025. We organize the review into two broad dimensions:

(a) Explainable AI in Software Engineering (XAI4SE).

(b) Ethical AI frameworks and their translation into concrete tools or practices.

2.1 Explainable AI in Software Engineering (XAI4SE)

Explainable Artificial Intelligence (XAI) is a crucial aspect of software engineering (SE) striving to enhance the transparency, interpretability, and trustworthiness of AI-driven tools. In software engineering applications, XAI allows for gaining insights into automated predictions and recommendations, aiding in debugging, validation, and decision-making processes [17] [21]. As AI-driven tools and methodologies become integral to defect prediction, testing, and maintenance, explainability has emerged as a critical aspect for ensuring successful interactions between humans and AI.

There are a few systematic literature reviews available that give an overview of the use of XAI in SE. Mohammad Khani et al. performed a wide systematic literature review of over 800 studies and selected 24 studies that are highly relevant to the field of XAI in software engineering [21]. Their findings indicate that most of the XAI systems are used for software maintenance, software defect prediction and software testing. The most popular post-hoc explanation methods are LIME and SHAP. The study also shows that there are considerable differences in the metrics used for the evaluation of explanation quality that hinder reproducibility and cross-study comparison [21], [28].

Likewise, Arora et al. conducted a survey of XAI techniques for the software development lifecycle (SDLC) in a phase specific manner [3]. From their analysis, it is understood that research is heavily focused on late SDLC phases like testing and maintenance and early SDLC phases like requirements engineering and design are under-researched. The authors distinguished four types of explanation techniques: rule-based, attention-based, counterfactual, and example-based, with a particular focus on the fact that types of

explanations should be carefully matched to the features of specific SE tasks [3].

There are no uniform methods for evaluating explanations, and this has led to motivation for research on the reliability and consistency of explanations. Awal and Roy introduced a framework called “EvaluateXAI” to evaluate the reliability and consistency of rule-based XAI techniques for software analytics tasks [5]. Their findings show that there is a large variation in the quality of explanations across models and datasets, which highlights the importance of having both standardized and purpose-specific evaluation metrics to facilitate the real deployment of XAI techniques in industrial SE environments.

Empirical studies with practitioners have been conducted to examine human-centered explainability. Huang et al. explored XAI explanations in the code smell prioritization domain and showed that explanation alignment with developer's expectations has a major impact on usability and trust [15]. Similarly, Jiarpakdee et al. investigated practitioners' perceptions of visual explanations regarding defect prediction models, which revealed that the developers preferred concise, actionable explanation that was directly associated with software artifacts and not a statistical output [16]. These results show that there is a consistent difference between technically correct explanations and explanations that are effective for supporting real world development tasks.

In addition, the role of explainability in enhancing the collaboration between developers and AI has also been the subject of research. Existing studies show that XAI enhances the interactive debugging of AI systems, helps in iterative refinements of AI tools, and raises software maintainability by striking a balance between automation and human knowledge [8]. The study of explainable defect prediction also highlights the importance of helping developers to take more informed corrective actions and gain greater trust in the model, thanks to its ability to provide interpretable explanations [2], [27].

Overall, the literature identified through XAI4SE shows that there are some significant trends. Even post hoc explanation methods like SHAP and LIME are the most popular methods so far, especially for defect prediction and maintenance tasks. However, some ongoing problems are the lack of coverage in the early SDLC phases, lack of sufficient human-centered design, and no consistent standards for the evaluation and explanation of the SDLC phases [3], [7], [17], [21]. These limitations highlight the need for improvements in the real-world application of XAI in software engineering processes.

2.2 Ethical AI Frameworks & Their Application in AI Systems

Ethical Artificial Intelligence (AI) in software development revolves around the principles of fairness, accountability, transparency, privacy, and social responsibility in AI systems. With the growing use of AI models in software engineering, including code review, defect prioritization, automated maintenance, and deployment decisions, ethical issues are becoming more vital to avoid biases, unsafe practices, and unintended consequences [1], [9], [10]. Ethical AI frameworks offer broad principles and guidelines designed to direct the development, design, and deployment of AI systems in a responsible manner within software development contexts.

The idea of ethical AI goes beyond performance to human values and societal norms, as several studies highlight. Abbas et al. discussed the development of AI systems that are fair and equitable by using techniques that are model interpretability and transparency, showing that the design of ethical AI frameworks should include the development of fairness-aware algorithms, bias detection mechanisms, and decision reporting transparency to ensure the trustworthiness of AI systems [1]. Empirical evidence also shows that failing to address fairness violations in AI-powered software analytics can have negative impacts on the quality of the decisions being made and the outcomes of the developers [9].

Explainability is a key component of the implementation of ethical principles in AI. McDermid et al. discussed technical and ethical aspects of explainability of AI models, emphasizing that explainable models enable accountability, auditability, and regulatory compliance in AI systems [20]. Explainable AI unfolds by rendering the model's behavior transparent and explaining why it makes its decisions, which can reduce ethical risks stemming from opaque decision-making processes, especially in critical and high-impact software engineering applications [15] [20]. Although there are frameworks on ethics in AI, several systematic reviews have pointed to the lack of synergy between ethical principles and implementation. Khan et al. [17] and Prem et al. [10] note that while frameworks often include the elements of fairness, transparency, and accountability, few include specific methods, tools, or metrics that can be easily applied to software development processes. Without a viable framework to tools gap, ethical guidelines are not yet effective for the development and deployment of AI systems in real-world scenarios.

In recent years, research has been carried out on how ethical frameworks for AI can be transformed into operational practice. Research has been conducted recently on turning ethical AI concepts into operational practice. Hort et al. have developed search-based automatic repair techniques that combine fairness and accuracy concerns in the decision-making process of software systems [14] and showed how these ethical goals can be integrated into the algorithmic optimization process directly. Ortega-Bolaños et al. summarized the tools for evaluating ethical compliance of AI systems and highlighted that a robust governance framework with continuous monitoring, auditing, and risk assessment is crucial for the responsible use of AI [23].

The ethical risks are especially significant in the realm of generative AI and large language model-based software development. Tuomisto outlined several ethical issues that can arise from generative AI tools, such as biased suggestions for code, intellectual property rights infringement, and misuse, and provided some recommendations to prevent these ethical problems, including automated algorithms for detecting bias, policy implementation, and human oversight [31]. Building on this, Weidinger et al. discussed the nature of societal and operational risks associated with LLM, including their capacity to perpetuate existing biases, misinformation, and discriminatory behavior, highlighting the need to have ethical governance across the lifespan of the AI system [33].

Rajuroy also explored ethical issues of AI in software engineering, focusing on the ethical dilemma surrounding the automation of software tasks and its implications for ethical standards [24]. The research highlights the need for the iterative introduction of ethical safeguards, accountability measures, and transparency protocols into AI-driven software development workflows, further underscoring the potential for AI to perpetuate bias, inaccuracies, and unsafe coding practices.

The literature shows that ethical AI frameworks serve as essential steppingstones for developing AI systems responsibly, but their utility in tangible tools, metrics, and practices for software engineering workflows is dependent on their successful translation [1], [9], [10], [14], [20], [23], [24], [31], [33]. Combining ethical principles with explainable AI techniques provides a viable approach for making ethics more practical and concrete in software engineering, while also increasing transparency and building trust in AI systems.

Table 1. Summary of Selected Literature on Explainable and Ethical AI in Software Engineering (2020–2025)

S • N o	Reference	Year	Focus / Domain	Methodology / Technique	SDLC Phase / Context	Key Findings
1	Mohammadkhani et al. [21]	2023	XAI in SE	Systematic Literature Review (800+ studies)	Maintenance , defect prediction, testing	Post-hoc explanation methods like LIME/SHAP widely used; evaluation metrics inconsistent
2	Arora et al. [3]	2025	Phase-specific XAI	Rule-based, attention-based, counterfactual, example-based	All SDLC phases	Early SDLC underexplored; alignment of explanation type with task required
3	Awal & Roy [5]	2024	XAI reliability	EvaluateXAI framework	Software analytics	Explanation reliability varies across tasks/models; standardized evaluation needed
4	Huang et al. [15]	2024	Developer-aligned XAI	Case study on code smell prioritization	Maintenance	Aligning explanations with developers improves usability and trust
5	Jiarpakdee et al. [16]	2021	Visual explanation perception	Survey / Practitioner feedback	Defect prediction	Explanations must be actionable, concise, and task-aligned
6	Shah [8]	2024	Developer-AI collaboration	Experimental study	Debugging, iterative improvement	XAI mediates between human expertise and AI automation, promotes trust
7	Khan et al. [17]	2023	XAI systematic review	Literature review	SE lifecycle	Gaps in unified standards, early SDLC coverage, human-centered evaluation
8	Abbas et al. [1]	2023	Ethical AI	Algorithmic fairness, interpretability	Model deployment	Fairness-aware algorithms and transparent decision reporting increase trust
9	Ferrara et al. [9]	2023	Ethical AI	Fairness-aware ML engineering	SE workflows	Measurement and improvement of fairness in ML models are critical

10	McDermid et al. [20]	2020	Technical & ethical XAI	Conceptual analysis	Safety-critical applications	Explainable models enhance accountability, trust, and compliance
11	Prem et al. [10]	2023	Ethical AI frameworks	Systematic review	SE workflows	Gap exists between ethical principles and practical integration
12	Hort et al. [14]	2024	Fairness in software repair	Search-based repair techniques	Decision-making software	Automated repair can reduce bias and improve fairness
13	Ortega-Bolaños et al. [23]	2024	Ethical AI tools	Systematic review	SE tools	Governance, monitoring, and auditing necessary for ethical AI adoption
14	Tuomisto [31]	n.d.	Generative AI ethics	Conceptual analysis	Software development	Bias detection, policy enforcement recommended for responsible adoption
15	Weidinger et al. [33]	2021	Social risks of AI	Conceptual / case examples	Large language models	Misinformation, discrimination, societal biases; operational ethical risks
16	Rajuroy [24]	2025	AI ethics challenges	Analytical study	SE pipelines	Integration of oversight and governance critical to avoid biased outputs
17	Al-Smadi et al. [2]	2023	XAI for defect prediction	Shapley values	Testing	Reliable prediction of defects with interpretable ML models
18	Cao et al. [7]	2025	XAI in SE	Systematic literature review	Software engineering	Need for interpretable ML in SE tasks; gaps in evaluation metrics
19	Schäfer et al. [26]	2023	LLMs for unit tests	Empirical evaluation	Testing	LLM-generated unit tests are feasible; explainability improves developer trust
20	Schulte et al. [27]	2024	XAI explanations for bugs	LIME & SHAP	Bug prediction	Explanations improve understanding; alignment with developer expectations needed
21	Shin et al. [28]	2021	Software defect XAI	Survey	Testing & maintenance	XAI adoption still limited; evaluation standards lacking
22	Li et al. [19]	2022	Data-driven XAI	Survey	SE tools	Knowledge-aware XAI improves interpretability and model understanding

23	Mustaqeem et al. [22]	2024	Transparent defect prediction	Hybrid model SPAM-XAI	Testing	Hybrid models enhance reliability and interpretability
24	Tanaka et al. [29]	2024	Generative AI risk assessment	Taxonomy	Risk assessment	Ethical evaluation required for AI-generated software artifacts
25	Vaithilingam et al. [32]	2022	Usability of code generation	User study	Code generation	Human-centered evaluation crucial for LLM-based tools
26	Giunti & Doherty [12]	2024	Generative AI in review process	Design science research	Automated review	Co-creation with AI enhances systematic review processes
27	Haque & Li [13]	2023	ChatGPT in debugging	Empirical analysis	Maintenance	AI tools can assist debugging; explainable outputs increase trust
28	Asare et al. [4]	2024	AI-assisted coding risks	Comparative study	Code generation	AI-generated code can introduce vulnerabilities similar to humans
29	Arora et al. [3]	2025	Lifecycle XAI survey	Phase-specific analysis	SDLC	Emphasizes need for phase-aligned XAI approaches
30	Bridging the Gap [6]	n.d.	XAI in defect detection	Preprint survey	Testing	Highlights interpretable ML techniques for defect detection
31	Explainable AI in SE [8]	n.d.	Developer-AI collaboration	Conceptual & survey	All SDLC phases	XAI improves collaboration and decision-making
32	Salem et al. [25]	2025	CI/CD automation with AI	Case study	Deployment	LLMs can automate code review; ethical considerations important
33	Kumari & Das [18]	2024	Relevant comment generation	AI-based system	Code documentation	Effective XAI improves comment relevance and developer understanding

3. Research Methodology

In this review paper, the authors took a structured and transparent approach to identify, select and synthesize research work related to Explainable Artificial Intelligence (XAI) in the period of 2020–2025 and in the field of Ethical AI (EAI)

applications in Software Engineering (SE). This methodology follows standard guidelines for the systematic and semi-systematic literature reviews with the principles and themes of PRISMA and thematic synthesis. This methodology is intended

to provide the accuracy, reliability and consistency of the review process.

This is a review paper that follows a structured and transparent methodology for identification, selection and synthesis of research related to Explainable Artificial Intelligence (XAI) and Ethical AI applications in Software Engineering (SE) from 2020 to 2025. The approach followed in this methodology is guided by well-established methods followed in systematic and semi-systematic literature reviews, such as PRISMA guidelines and thematic synthesis. This methodology is aimed at ensuring accuracy, reliability and consistency in the review process.

3.1 Search Strategy

A wide variety of scholarly databases and platforms were searched to obtain a comprehensive sample of relevant research, such as ResearchGate, IEEE Xplore, ACM Digital Library, Springerlink, ScienceDirect and arXiv preprint repository. The following keywords and boolean combinations were chosen: “Explainable AI in Software Engineering,” “XAI for software tools,” “XAI for SDLC phases,” “Explainable machine learning in SE,” “Ethical AI in software engineering,” “Responsible AI for software tools,” “AI ethics frameworks AND software systems,” “Transparency AND AI tools for developers,” “Bias/Fairness in AI-driven code generation”, and “Ethics of AI-based software development”. The following search terms were chosen to include publications on both technical explainability and ethical aspects of software engineering processes.

3.2 Inclusion Criteria

Cross-disciplinary relevance was ensured by including only studies from the software engineering domain, which focused on software engineering tools, developer-centered AI systems, software engineering workflows, software development life cycle (SDLC) phases, or AI quality/process frameworks. For relevance, only studies that met the following inclusion criteria were included: studies focused on software engineering tools, developer-centered AI systems,

software engineering workflows, software development life cycle (SDLC) phases, or AI quality/process frameworks, publication year (2020-2025), only peer-reviewed journal articles, conference papers, high-quality preprints (arXiv), systematic reviews, and industry/reputable organization whitepapers (IEEE, IBM, Google), and only English-language studies.

3.3 Exclusion Criteria

Papers with no SE context, those that were focused solely on core machine learning, ethics in healthcare, robotics, medicine, military AI, and automotive systems were excluded; papers that discussed XAI as a general ML tool that is not connected with developers or SE practices were also excluded; papers that were blogs, survey reports, and/or non-scholarly essays were also excluded.

3.4 Study Selection Process

The review underwent three stages of screening. The first search yielded 284 papers through keyword searching the databases. After duplicate removal, 176 papers remained after title and abstract screening; and 74 papers were screened after title and abstract removal. After the detailed reading and relevance evaluation in the assessment stage of full text, 38 papers were selected for the final review. These encompassed papers on XAI for SE, SE phase-specific papers on XAI, papers on ethical frameworks for AI, developer studies on the utility of explanations, and industry guidelines (IEEE 7000-series, IBM, Google AI Ethics).

3.5 Data Extraction

The data extracted for each included study included: authors, year and venue; problem addressed; XAI technique or ethical framework employed; the software engineering phase addressed; the type of tool or model (LLM, rule-based, code generation assistant, prediction model); evaluation metrics (accuracy, interpretability, usability, fairness); and key findings and limitations. The data obtained was

then transferred to comparative tables to help identify patterns.

3.6 Categorization and Thematic Analysis

The selected papers were classified based on the two main dimensions in this review: Explainable AI (XAI) and Ethical AI (Ethical) in Software Engineering. The two-dimensional categorization was chosen to systematically examine technical explainability features and ethical considerations of AI-powered SW engineering tools. Thematic grouping was used within each dimension to identify the major research trends, the methodological patterns, and the research gaps.

3.6.1 Review and Thematic Analysis Protocol

A rigorous, transparent and reproducible categorization and thematic analysis process were carried out by using a structured qualitative review protocol. The following steps were followed in the protocol. Selection and screening of papers: Papers published between 2020 and 2025 that were found relevant were selected and screened for explainable AI and ethical AI in software engineering, with papers that did not have an empirical or conceptual contribution related to AI-driven software engineering tools omitted. Second, main categorization: studies were first categorized into two main categories: "Explainable AI in Software Engineering (XAI4SE)" and "Ethical AI Frameworks and Their Application in AI Systems" with studies covering both categories analyzed across the different categories to capture interdependencies. Third, thematic coding and grouping: recurring concepts about techniques, applications, evaluation methods, risks and design

principles were identified through open coding and were grouped into higher-level themes within each dimension through axial coding, creating coherent thematic groups within each dimension. Fourth, thematic synthesis: a qualitative synthesis was performed in order to identify patterns and key themes in the study of XAI and the study of AI ethics, strengths and weaknesses of current approaches to practical software engineering applications and under-researched areas and gaps that need further research. Fifth, cross-dimensional analysis: Interconnections between explainability and ethical issues were explored to gain insight into the role of XAI techniques in supporting ethical principles such as transparency, accountability, fairness, and trust in AI-based software systems.

3.6.2 Thematic Categorization

For the Explainable AI in Software Engineering dimension, studies were categorized into themes related to the use of XAI techniques in software engineering tools (XAI4SE), integration of XAI throughout various stages of the Software Development Lifecycle (SDLC), developer-centric assessments and practitioner views on AI explanations, and reliability, consistency, and assessment of XAI frameworks.

In the ethical AI in Software Engineering dimension, thematic grouping focused on ethics in AI frameworks and guidelines, and their implementation in SE tools, ethical risks (bias, privacy issues, security threats), ethical evaluation of tools from a human perspective, and ethical aspects throughout SDLC stages.

Table 2. Thematic Analysis of Explainable and Ethical AI in Software Engineering

Theme	Sub-Theme	Key Insights	Representative References
Explainable AI in Software Engineering (XAI4SE)	Post-hoc explanation methods	LIME, SHAP, rule-based, example-based approaches are widely used for defect prediction, code review, and automated debugging.	[2], [5], [15], [27], [28]
	Phase-specific SDLC coverage	Early SDLC phases (requirements, design) are underexplored; maintenance and testing dominate.	[3], [7], [21]

	Developer-centered evaluation	Developers prefer actionable, concise, and interpretable explanations aligned with real tasks.	[15], [16], [8], [12]
	XAI frameworks and reliability	EvaluateXAI framework highlights variability in explanation reliability; standard metrics are lacking.	[5], [17], [21]
Ethical AI in Software Engineering	Ethical frameworks & operationalization	Fairness, accountability, transparency are emphasized, but practical enforcement in SE tools is limited.	[1], [9], [23], [24], [33]
	Risk assessment & mitigation	AI systems can introduce bias, privacy risks, security vulnerabilities; operational safeguards needed.	[4], [10], [31], [33]
	Human-centered evaluation	Alignment of ethical safeguards with developer workflows increases adoption and trust.	[1], [2], [24], [33]
	SDLC integration	Ethical AI considerations are often neglected in early SDLC phases, where bias mitigation is most effective.	[3], [7], [23]
Cross-Dimensional Observations	Explainability & ethical alignment	XAI supports ethical principles by improving transparency, accountability, and auditability of AI outputs.	[1], [2], [15], [27]

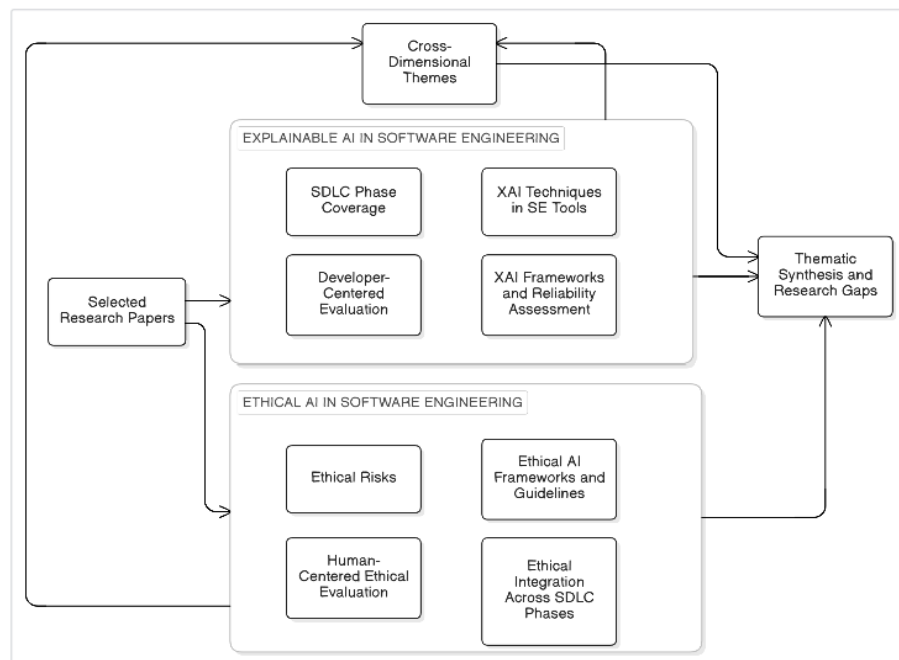


Figure 1. Two-Dimensional Categorization and Thematic Analysis Framework for Explainable and Ethical AI in Software Engineering

4. Results

This section shares the results of the qualitative synthesis and thematic analysis of the selected literature on Explainable Artificial Intelligence (XAI) and Ethical AI in Software Engineering (SE). The results are presented in two main dimensions of this review, namely AI in Software Engineering – Explainable and Ethical AI in Software Engineering, with the identification of dominant trends, strengths, weakness and research gaps from the latest studies.

4.1 Explainable AI - Results in Software Engineering

The analysis shows that explainable AI techniques are becoming more common in software engineering tools, deserving of greater transparency, trust, and understanding of AI outputs. Most existing studies have been dedicated to explaining the results of the applied classifiers, like feature attribution, rule-based explanations, and example-based reasoning, to facilitate the following tasks such as defect prediction, code quality assessment, test prioritization, and maintenance.

The trend is evident towards an application focus on maintenance and tests, and the initial stages of SDLC are comparatively under-explored, including requirements engineering and software design. Multiple studies have shown that explainability increases the developers' trust in the AI recommendations, with concise, task-relevant explanations and integration with real-world development workflows being the most important features. It is reported that a combination of visual and interactive explanations is more effective than a purely technical approach.

Note, however, that there are also significant weaknesses in the results. Studies frequently use ad hoc or inconsistent measures to assess quality of explanations, which complicates comparison between studies. Numerous explanation methods focus on transparency of the model rather than on its actual use, and consequently on explanations that are technically correct but not easily understood or acted upon by the developer. The

results indicate the differences between researching explainability and its successful use in the real software engineering setting.

4.2 Results for Ethical AI in Software Engineering

The findings about ethical AI highlight the increasing awareness of F.A.T. and responsibility as key criteria for AI-powered tools in software engineering. While there are plenty of studies on ethical frameworks, guidelines, and high-level principles, there has been a growing interest in the potential risks of bias propagation, lack of accountability, intellectual property issues, and misuse of generative AI systems.

The most frequently encountered level of addressing ethical consideration is at a conceptual or governance level, and there is limited integration of ethical issues into software engineering pipelines. There are few studies with specific mechanisms or specific tools that offer the opportunity to incorporate ethical safeguards into day-to-day development work. Ethical AI is always closely tied to XAI, as human oversight, auditability, and explainability are always cited as key enablers for ethical compliance.

It is also found that ethical risks are more likely to be recognized after deployment, as opposed to being addressed in the initial stages of the SDLC. Such a response-based strategy makes ethical frameworks ineffective and potentially leads to skewed and damaging results. The outcomes confirm the importance of developer-oriented ethical tools that can assist in making ethical decisions throughout the design, construction and testing phases.

4.3 Cross-Dimensional Findings and Gaps

Overall, the findings suggest that there is a deep connection between explainability and ethical AI, both in terms of their dimensions and the extent to which they are interdependent. Explainable AI is often seen as a key element in ensuring that AI systems are accountable, transparent, and trustworthy. In any case, there is still no common evaluation practice for either dimension, and early

SDLC phases are covered minimally with both of them, and they focus on developer usability to a limited degree.

The overall findings indicate that, although significant strides have been achieved in the conceptualization of explainable and ethical AI in the software engineering context, its

implementation is still fragmented. The literature suggests that a need exists for integrated, developer-focused, and lifecycle-aware methods for providing explainability and ethical protection to enable trustworthy and responsible AI-powered software engineering.

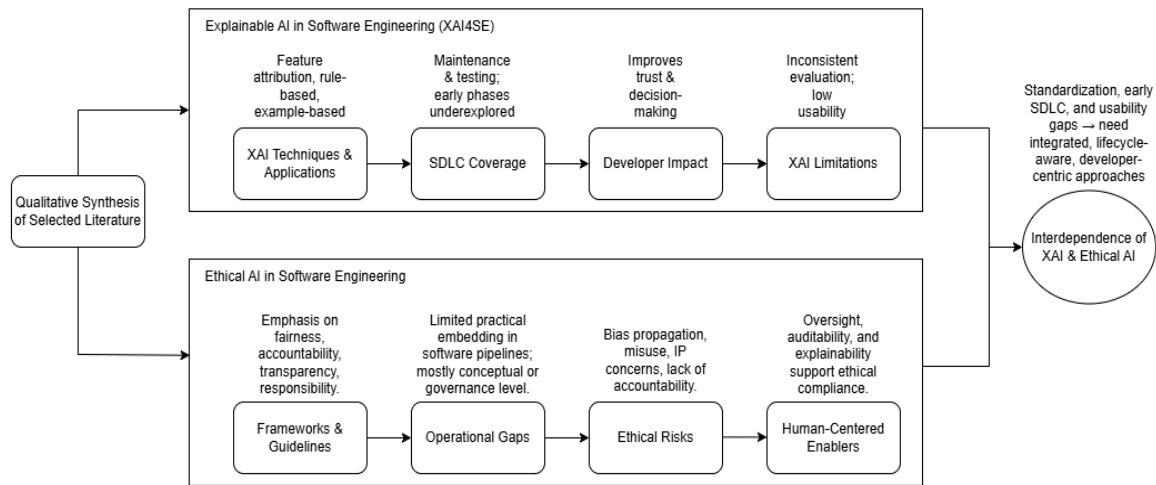


Figure 2. Results from the Literature on Explainable and Ethical AI in Software Engineering

Table 3. Mapping of Review Results to Research Questions

Research Question	Focus Area	Key Results Identified from the Review	Implications for Software Engineering
RQ1: Explainability Dimension	Effectiveness of XAI techniques in SE tools	XAI techniques improve understanding and trust when explanations are task-specific, concise, and aligned with developer workflows. Post-hoc methods (e.g., SHAP, LIME) are widely used, but explanation reliability and usability vary across tools and tasks.	There is a need for developer-centered XAI designs and standardized evaluation metrics to ensure practical usefulness in real-world SE environments.
RQ2: Ethical AI Dimension	Practical implementation of ethical AI frameworks	Ethical principles such as fairness, accountability, and transparency are well-established in theory but remain weakly operationalized in SE tools. Bias mitigation, accountability mechanisms, and governance practices are inconsistently applied.	Bridging the gap between ethical frameworks and deployable SE tools requires integrating ethical safeguards into AI pipelines and development workflows.
RQ3: Integration & SDLC Gaps	Coverage of XAI and ethical AI	Most explainable and ethical AI research concentrates on maintenance and testing phases,	Early integration of explainability and ethics can prevent downstream

	across SDLC phases	while early SDLC stages such as requirements engineering and design are significantly underrepresented.	bias, improve decision quality, and enhance long-term software trustworthiness.
Cross-RQ Insight	Relationship between XAI and Ethical AI	Explainability acts as a key enabler for ethical AI by supporting transparency, accountability, and auditability of AI-driven SE tools.	Joint consideration of explainability and ethics is essential for building trustworthy, human-centered AI-enabled software systems.

5. Discussions

The literature review results are presented with respect to the formulated research questions covering the two main aspects of this research: Explainable Artificial Intelligence (XAI) and Ethical AI in Software Engineering. The key points of insights, common challenges, and gaps in knowledge from recent studies are emphasized.

5.1 Explainability in Software Engineering (RQ1)

The literature reviewed highlights the importance of explainable AI techniques for enhancing the interpretability and trust of AI-powered software engineering tools. Assist tasks like defect prediction, code quality assessment, test prioritization and maintenance activities are assisted through various explanation methods such as SHAP, LIME, rule-based models, and example-based explanations. These methods allow software engineers to know what features are influential, check model outputs and troubleshoot wrong or unexpected predictions.

Results demonstrate, however, that while explaining technical models is important for their usefulness, it alone is not enough. There has been some research that suggests that the main factor that influences the effectiveness of the explanation is alignment with software engineering tasks and developers' expectations. While explanations might be technically correct, it is not likely to be supportive of action in decision making if they are overly detailed, wordy, and/or not directly related to actual development practices. This highlights

the historical gap between algorithmic explainability and human interpretability.

Another major problem that is noted is the lack of uniform scales to evaluate the quality of explanations. Usability, cognitive load and actionability are less commonly measured in current evaluations and model fidelity or accuracy-based measures are more common. Comparing studies and tools is challenging, however, because of this. Based on these results, we recommend that future work in XAI in software engineering focuses on developer-focused evaluation approaches and task-specific explanation designs to boost adoption in practice.

5.2 Ethical AI in Software Engineering (RQ2)

There is a growing awareness of the need for ethical AI in AI-enabled software engineering systems, as evidenced by the literature. Fairness, accountability, transparency and risk mitigation are among the ethical principles which are discussed in a wide range of conceptual approaches and policy-oriented investigations. The principles are designed to help prevent bias, promote responsible automation, and foster trust in AI-powered software tools.

Despite all this theory, the review shows a huge disconnect between the ethical frameworks for AI and their application in practice. Ethical safeguards are generally summarized in a general way, and there is little guidance on how these can be put into practice in software engineering pipelines. Many AI tools, therefore, do not have systematic processes in place to identify, hold accountable for, or monitor ethical compliance.

Furthermore, the ethical concerns with the use of AI in software engineering tools have not been adequately researched in empirical studies. While there are documented acknowledgements of issues like biased code recommendations, vulnerability propagation, intellectual property concerns and over-reliance on automated decision making, it is not consistently addressed with concrete technical solutions. The results highlight the importance of practical, ethical AI tools, frameworks, and assessment practices to be embedded in the actual software development lifecycle.

5.3 Integration Across the Software Development Lifecycle (RQ3)

Ethical AI research throughout the Software Development Lifecycle. Existing research is mostly concentrated on the later phases of the SDLC, such as software maintenance, defect prediction and software testing. Historical data and measurable outcomes are advantageous for these phases because they allow for easier analysis and evaluation using AI.

On the contrary, attention is given to only a few early SDLC phases (requirements engineering, system design and architectural decision making). Without explainability and ethical concerns at these points, there is a heightened risk of introducing bias, design flaws, or hidden logic for decisions early in the development process. Filling these gaps could enhance the quality of software in the long term and mitigate downstream risks.

The results further indicate that explainability is key as an enabler for ethical AI throughout the SDLC. The transparency and interpretability of models foster accountability, auditability and regulatory compliance, translating ethical principles to practice. Nevertheless, existing studies have limited their scope by failing to provide common frameworks that collectively address explainability and ethics throughout the SDLC.

6. Conclusion

In this review, we conducted an analysis on recent literature for Explainable Artificial Intelligence

(XAI) and Ethical AI in Software Engineering (SE) from the year 2020 to 2025. The analysis was organized across two main dimensions of Explainable Artificial Intelligence in Software Engineering (XAI4SE) and ethical AI frameworks and how they can be transformed into tools and practices. The purpose of the review was to uncover prevailing research trends, practical challenges, and open research questions in the realm of AI responsible integration into software engineering workflows.

The results indicate that the use of XAI techniques in software engineering is growing and is applied in various areas such as defect prediction, software testing, software maintenance and software quality assessment. The most popular methods are still post-hoc explanation methods. These techniques increase transparency and developer trust but are not standardized for evaluation, and are not uniform throughout the Software Development Lifecycle (SDLC). Limited attention is still paid to early SDLC phases, like requirements engineering and design, and many explanations are still hard to interpret and implement in practice.

In tandem, ethical AI studies focus on values like fairness, accountability, transparency, and responsibility. The majority of ethical frameworks are still abstract, and few of them are applied in practice to software engineering tools. Ethics issues are usually communicated as a response to an incident after a deployment takes place, rather than being anticipated and minimised in early stages of development.

An important outcome of this review is the recognition of the strong interdependence between explainable and ethical AI. Explainability is one of the essential pillars for responsible AI, fostering accountability, auditability and trust. However, these two dimensions have many common problems such as the lack of a common set of evaluation metrics, poor early SDLC integration, and a narrow attention to developer-centered usability.

Overall, this review shows that the adoption of integrated, lifecycle-aware and developer-centric strategies incorporating explainability and ethical

protection is required. Overcoming these challenges is key to the development of responsible and trustworthy software engineering practices with AI.

7. REFERENCES

- [1] S. Abbas *et al.*, “Ensuring fairness and equity in AI systems using algorithms for model interpretability and transparency,” *Journal of Computational Analysis and Applications*, vol. 31, no. 3, pp. 517–531, 2023.
- [2] Y. Al-Smadi *et al.*, “Reliable prediction of software defects using Shapley interpretable machine learning models,” *Egyptian Informatics Journal*, vol. 24, no. 3, p. 100386, 2023, doi: 10.1016/j.eij.2023.05.011.
- [3] L. Arora *et al.*, “Explainable artificial intelligence techniques for the software development lifecycle: A phase-specific survey,” *arXiv preprint arXiv:2501.03210*, 2025.
- [4] O. Asare, M. Nagappan, and N. Asokan, “Is GitHub Copilot as bad as humans at introducing vulnerabilities in code?” *arXiv preprint arXiv:2204.04741*, 2024.
- [5] M. A. Awal and C. K. Roy, “EvaluateXAI: A framework to evaluate the reliability and consistency of rule-based XAI techniques for software analytics tasks,” *arXiv preprint arXiv:2407.13902*, 2024.
- [6] Anonymous, “Bridging the gap: Exploring explainable AI for interpretable machine learning models in software defect detection,” *EasyChair Preprint*, 2023.
- [7] S. Cao *et al.*, “A systematic literature review on explainability for machine and deep learning-based software engineering research,” *arXiv preprint arXiv:2401.14617*, 2025.
- [8] Anonymous, “Explainable AI in software engineering: Enhancing developer-AI collaboration,” *American Journal of Engineering and Technology*, 2023.
- [9] C. Ferrara *et al.*, “Fairness-aware machine learning engineering: How far are we?” *Empirical Software Engineering*, vol. 29, no. 1, p. 9, 2023, doi: 10.1007/s10664-023-10402-y.
- [10] Anonymous, “From ethical AI frameworks to tools: A review of approaches,” *AI and Ethics*, 2023, doi: 10.1007/s43681-023-00258-9.
- [11] B. Gezici Geçer, “Advancing software defect prediction through ensemble XAI methods: Insights and performance evaluation,” M.S. thesis, 2024. [Online]. Available: <https://hdl.handle.net/11655/35990>
- [12] G. Giunti and C. P. Doherty, “Cocreating an automated mHealth apps systematic review process with generative AI: A design science research approach,” *JMIR Medical Education*, vol. 10, p. e48949, 2024, doi: 10.2196/48949.
- [13] M. A. Haque and S. Li, “The potential use of ChatGPT for debugging and bug fixing,” *EAI Endorsed Transactions on AI and Robotics*, vol. 2, no. 1, 2023, doi: 10.4108/airo.v2i1.3276.
- [14] M. Hort *et al.*, “Search-based automatic repair for fairness and accuracy in decision-making software,” *Empirical Software Engineering*, vol. 29, no. 1, p. 36, 2024, doi: 10.1007/s10664-023-10419-3.
- [15] Z. Huang *et al.*, “Aligning XAI explanations with software developers’ expectations: A case study with code smell prioritization,” *Expert Systems with Applications*, vol. 238, p. 121640, 2024, doi: 10.1016/j.eswa.2023.121640.
- [16] J. Jiarpakdee, C. Tantithamthavorn, and J. Grundy, “Practitioners’ perceptions of the goals and visual explanations of defect prediction models,” *arXiv preprint arXiv:2102.12007*, 2021.
- [17] A. Khan *et al.*, “A systematic literature review of explainable artificial intelligence in software engineering,” *arXiv preprint arXiv:2302.06065*, 2023.

- [18] T. Kumari and A. Das, "Towards development of an effective AI-based system for relevant comment generation," in *Proc. ACM Int. Conf. on Software Engineering and Knowledge Engineering*, 2024, doi: 10.1145/3632754.3632770.
- [19] X.-H. Li et al., "A survey of data-driven and knowledge-aware explainable AI," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 1, pp. 29–49, 2022, doi: 10.1109/TKDE.2020.2983930.
- [20] J. A. McDermid et al., "Artificial intelligence explainability: The technical and ethical dimensions," *Philosophical Transactions of the Royal Society A*, vol. 379, no. 2207, p. 20200363, 2020, doi: 10.1098/rsta.2020.0363.
- [21] A. H. Mohammadkhani et al., "A systematic literature review of explainable AI for software engineering," *arXiv preprint arXiv:2302.06065*, 2023.
- [22] M. Mustaqeem et al., "A trustworthy hybrid model for transparent software defect prediction: SPAM-XAI," *PLOS ONE*, vol. 19, no. 7, p. e0307112, 2024, doi: 10.1371/journal.pone.0307112.
- [23] R. Ortega-Bolaños et al., "Applying the ethics of AI: A systematic review of tools for developing and assessing AI-based systems," *Artificial Intelligence Review*, vol. 57, no. 5, p. 110, 2024, doi: 10.1007/s10462-024-10740-3.
- [24] A. Rajuroy, "Ethical challenges in AI-driven software engineering: Striking the balance," unpublished manuscript, 2025.
- [25] D. O. Salem et al., "AI-driven continuous integration: Automating code review and deployment with large language models," in *Proc. 10th Int. Conf. on Fog and Mobile Edge Computing (FMEC)*, 2025, pp. 268–274, doi: 10.1109/FMEC65595.2025.11119365.
- [26] M. Schäfer et al., "An empirical evaluation of using large language models for automated unit test generation," *arXiv preprint arXiv:2302.06527*, 2023.
- [27] L. Schulte, B. Ledel, and S. Herbold, "Studying explanations for the automated prediction of bug and non-bug issues using LIME and SHAP," *Empirical Software Engineering*, vol. 29, no. 4, p. 93, 2024, doi: 10.1007/s10664-024-10469-1.
- [28] J. Shin et al., "Explainable software defect prediction: Are we there yet?" *arXiv preprint arXiv:2111.10901*, 2021.
- [29] H. Tanaka et al., "Taxonomy of generative AI applications for risk assessment," in *Proc. IEEE/ACM Int. Conf. on AI Engineering (CAIN)*, 2024, pp. 288–289.
- [30] Anonymous, "The role of explainable AI in automated software testing: Opportunities and challenges," *Preprints.org*, 2025.
- [31] T. Tuomisto, "Ethical considerations of using generative AI in software development," unpublished manuscript, 2023.
- [32] P. Vaithilingam, T. Zhang, and E. L. Glassman, "Expectation vs. experience: Evaluating the usability of code generation tools powered by large language models," in *Proc. CHI Conf. on Human Factors in Computing Systems*, 2022, pp. 1–7, doi: 10.1145/3491101.3519665.
- [33] L. Weidinger et al., "Ethical and social risks of harm from language models," *arXiv preprint arXiv:2112.04359*, 2021.