e-ISSN 2549-7472
Edumatic: Jurnal Pendidikan Informatika
Volume 10, Issue 1, **290-299.**
https://e-journal.hamzanwadi.ac.id/index.php/edumatic
DOI: 10.29408/edumatic.v10i1.34361

Hybrid Human AI SDLC for Rapid SaaS Development: Evidence from a 60 Days Case Study

Muhammad Hilmy Haidar Fawwazie [1], Nathan Daud [2] , Muhammad Iqbal Rabani [2], Budi Fajar Supriyanto [3, *]

[1] Universitas Gajayana, Indonesia
[2] Universitas Brawijaya, Indonesia
[3] Politeknik Negeri Jember, Indonesia
[*] Corresponding author: Budi Fajar Supriyanto, Politeknik Negeri Jember, Indonesia
✉ fajar.deroom@gmail.com

Received: 19 March 2026 | Revised: 2 April 2026 | Accepted: 26 April 2026 | Published: 29 April 2026

Abstract

There is a vacuum in the risk of architectural changes in critical systems because macro-architectural governance in AI-based software development is frequently ignored in current scholarly debate. The purpose of this study is to assess how well the Visualize, Integrate, Build, Execute (VIBE) architecture addresses the stability-speed contradiction in SaaS development. This study triangulated data from 465 automated CI/CD pipeline logs, 124 AI instruction tactic documentation records, and 42 test cases using comparative performance analysis and process tracking using an explanatory mixed-methods case study methodology on a stock market analytics platform. The study's key conclusions show a 50% boost in development efficiency, reducing a 60-day cycle to 30 days while preserving system reliability with an average latency of 1.2 seconds and a 99.9 percent availability rate. Specialist synergy was identified where humans became the primary cognitive players in architectural design at 90 percent, and AI as the executor of basic syntax at 85 percent. The research concludes that the architectural anchoring mechanism by humans is crucial for mitigating the risks of non-deterministic AI outputs. Theoretically, this study introduces the concept of human-AI cognitive alignment, while practically providing a validated roadmap for modernization of sensitive infrastructure such as Electronic Medical Records.

Keywords: hybrid software development lifecycle; human AI collaboration; AI assisted development; electronic medical records

To cite this article: Fawwazie, M. H. H., Daud, N., Muhammad Iqbal Rabani, & Supriyanto, B. F. (2026). Hybrid Human AI SDLC for Rapid SaaS Development: Evidence from a 60 Days Case Study. *Edumatic: Jurnal Pendidikan Informatika*, *10*(1), 290–299. https://doi.org/10.29408/edumatic.v10i1.34361

INTRODUCTION

The integration of Large Language Models (LLMs) into software engineering has redefined development practices by enabling AI-assisted code generation, automated debugging, and system prototyping. Empirical evidence indicates that LLMs significantly improve productivity at the micro-development level, particularly in tasks related to syntax generation and code completion (Wang & Wu, 2025; Romeo & Conti, 2026). This transformation has accelerated development workflows while simultaneously introducing structural challenges associated with system coherence and long-term maintainability.

In high-integrity domains such as financial analytics and Electronic Medical Records (EMR), system reliability and data precision constitute fundamental requirements. The increasing reliance on AI-generated code has raised concerns regarding the ability of engineering teams to preserve architectural consistency. Khan et al. (2024) and Yang et al. (2025) identify a growing tendency toward structural degradation when AI-generated outputs are not governed within a controlled framework. This condition is further intensified in high-concurrency environments where minor inconsistencies may propagate into system-wide failures. Consequently, the absence of rigorous governance mechanisms in AI-assisted development environments introduces significant risks to data integrity and system reliability (Kang & Park, 2025; Pesce & Cheungpasitporn, 2025).

The discourse on Software Development Life Cycle (SDLC) governance increasingly addresses the implications of AI-driven development, particularly the *stability–velocity paradox*, which reflects the tension between development acceleration and structural integrity. Hossain et al. (2025) and Esmaeilzadeh (2024) emphasize that AI adoption introduces governance challenges related to control, accountability, and system sustainability, especially in regulated environments. Although prior studies confirm the effectiveness of AI in improving micro-level development tasks, limitations emerge at the architectural level. Saravanos and Curinga (2023), and Leong et al. (2023) indicate that existing SDLC models lack mechanisms to manage non-deterministic AI outputs, while Arshad et al. (2025) highlight the need for scalable yet controlled development frameworks. In addition, Liu et al. (2026) demonstrate that hallucination in AI-generated code poses risks to system reliability. Collectively, the literature reveals an imbalance between efficiency-oriented advancements and the insufficient development of architectural governance mechanisms within AI-assisted software engineering

Current research on AI-assisted software development reveals several critical limitations. Empirical validation of mechanisms to control non-deterministic AI outputs in real-world development contexts remains limited, as existing studies predominantly emphasize security and access control without addressing architectural drift (Rahaman et al., 2023). In addition, integrated frameworks that combine deterministic architectural governance with AI-driven automation are not yet well established, particularly in managing hallucination risks in production systems (Liu et al., 2026). Furthermore, contextual evidence from emerging digital ecosystems, including Indonesia, is still underrepresented, indicating a lack of understanding of how human AI collaboration operates within diverse socio-technical environments (Kopuz & Kartal, 2025; Yitagesu et al., 2026)

A fundamental scientific limitation lies in the absence of a unified conceptual model that systematically aligns human decision-making with AI execution within the Software SDLC. Existing studies largely frame human AI interaction as task delegation, without establishing deterministic mechanisms to regulate AI behavior at the architectural level. Medvidović (2025) emphasizes the need for paradigm-level advancements that focus on systemic complexity rather than incremental improvements. In this context, the lack of controlled integration frameworks allows non-deterministic AI outputs to evolve without sufficient constraints, increasing the risk of structural inconsistency. This study is therefore justified by the need to develop a governance-oriented model in which human expertise functions as the primary architectural control, with financial analytics systems serving as a high-concurrency proxy to evaluate applicability in mission-critical domains such as healthcare

This study aims to evaluate the effectiveness of the Visualize, Integrate, Build, Execute (VIBE) framework in enhancing development efficiency while preserving architectural integrity within AI-assisted software development. It specifically investigates the role of human-led architectural control in constraining non-deterministic AI outputs through a 60-day empirical case study on a SaaS-based financial analytics platform. The novelty of this research lies in the introduction of a Human AI Cognitive Alignment model, which positions AI as a constrained executor operating within explicitly defined architectural boundaries. This approach extends existing perspectives on human AI collaboration by establishing a structured mechanism of architectural anchoring that ensures system-level coherence. The study contributes theoretically by formalizing architectural anchoring as a control mechanism against AI-induced structural drift, methodologically by integrating quantitative performance evaluation with qualitative process tracing in an explanatory mixed-methods design, and practically by delivering a validated development roadmap applicable to mission-critical systems requiring high data integrity, including EMR.

METHOD

This research employs an explanatory mixed methods case study to evaluate the effectiveness of the VIBE framework in maintaining structural integrity during accelerated development. VIBE comprises four strategic pillars: Visualize (establishing architectural blueprints), Integrate (harmonizing AI outputs with system logic), Build (constructing functional components with high-integrity standards), and Execute (deploying under constraints). This design is justified by the stability-velocity paradox, where AI-driven acceleration threatens the long-term coherence of mission-critical systems. By adopting an explanatory approach, the study identifies causal mechanisms specifically early-phase architectural anchoring that allow human developers to constrain non-deterministic AI outputs through phase-gated synchronization (Huang et al., 2025).

The ratio of the actual project time to a predefined professional baseline is the methodological definition of development efficiency. This baseline was constructed through expert estimation triangulation from three senior full-stack developers with over five years of specialized experience (Meske et al., 2025). Data collection triangulates three instruments: 465 automated logs from GitHub and Azure CI/CD pipelines, 124 manual process entries documenting prompting tactics in Cursor and Claude 3.5, and 42 black-box test cases. This multi-layered approach ensures that observed efficiencies reflect genuine structural quality. The development process is divided into four chronological phases over an eight-week cycle, as illustrated in Figure 1.

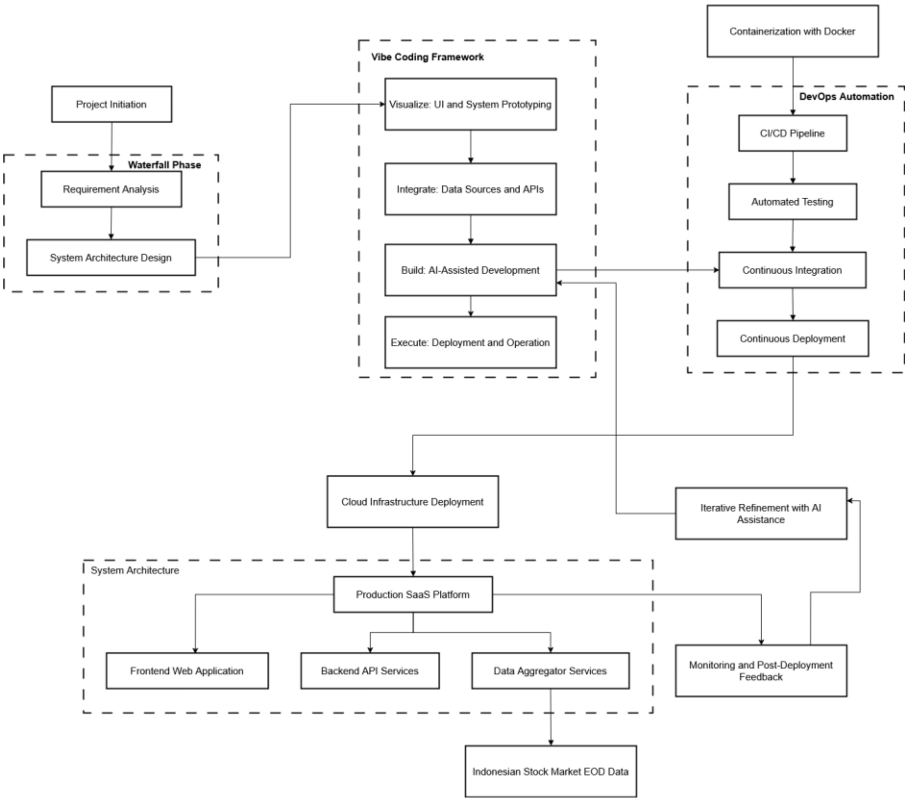


Figure 1. Hybrid sdlc + vibe coding workflow

Data analysis utilizes a two-tiered approach: comparative performance analysis for quantitative metrics and process tracing for qualitative interaction logs. This facilitates the statistical comparison of delivery timelines against the manual baseline while identifying critical handoff points in automated execution. The unit of analysis is the 60-day SaaS development cycle, with the research boundary limited to an Indonesian stock market analytics platform. This study offers a verified blueprint for updating delicate infrastructures like EMR without sacrificing data security by using public market data as a technical proxy.

RESULTS AND DISCUSSION

Results

The empirical results from applying the VIBE framework in a high-integrity system context are presented in this section. The evaluation examined operational data triangulated from continuous integration pipelines and system performance monitors across the compressed development cycle in order to thoroughly evaluate the framework's ability to overcome the stability-velocity contradiction. In addition to analyzing the crucial balance of human-AI collaboration, the examination focuses on how well the framework expedites the development timetable while retaining enterprise-grade reliability. Finally, Table 1 provides a thorough summary of the VIBE framework's quantitative evaluation and validation outcomes, including efficiency metrics, system stability, and architectural control distribution.

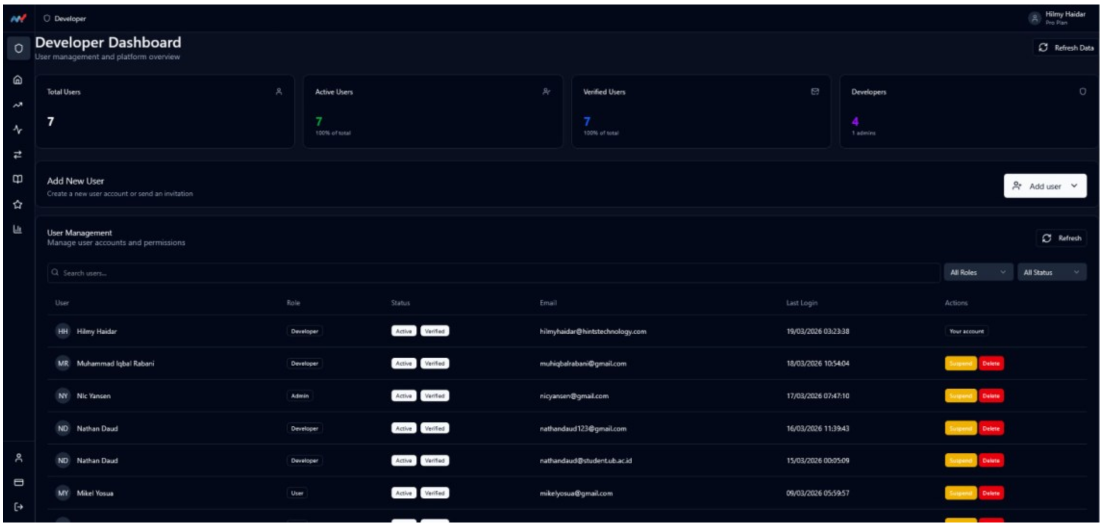
Table 1. Comprehensive results and validation of the vibe framework

Metric / Indicator	Result / Value	Validation & Descriptive Remarks
Visualise & Integrate Phase	60% Increase	Narrow 95% CI; triangulated from 465 CI/CD logs & 124 AI tactic records
Execute Phase	30% Increase	Lowest disparity; highlights the need for human deterministic accuracy in system hardening
System Reliability (Uptime)	99.9%	Empirically proves acceleration does not compromise technical integrity
Visualization Latency	1.2 seconds	Maintains stable enterprise-grade performance (seconds)

The Visualize & Integrate by Table 1 phase obtained the largest efficiency spike of 60%, according to the phase evaluation based on the metric information in the table. The precision level was highly supported by triangulating 465 CI/CD logs and 124 AI strategy records. The Execute phase, on the other hand, saw the least amount of efficiency increase (30%). This discrepancy essentially confirms how important human deterministic correctness is during the system hardening process. An uptime achievement of 99.9% and a stable visualization latency of 1.2 seconds show that, despite an acceleration in the development cycle, the stability of system performance has proven to remain at enterprise standards. This empirically shows that the AI-driven development speed does not compromise the technical integrity of the system.

Performance capability and functional was validated through high-concurrency stress testing, maintaining a 99.9% uptime over the 60-day observation period despite the accelerated delivery. Table 2's findings confirm that the expedited timeframe did not compromise technical integrity. Visual evidence from the resulting platform (Figures 2 and 3) serves as scientific validation of the system's capability to synchronize Level 3 market data in real-time, consistently achieving sub-second rendering delays averaging 1.2 seconds with a low of 0.8 seconds. Furthermore, the successful deployment of the Developer Dashboard (Figure 3) proves the framework's capacity to handle sensitive administrative infrastructures, specifically meeting complex requirements for Role-Based Access Control (RBAC) and secure identity governance. These results confirm that the compressed timeframe did not compromise technical integrity, as the backend integration successfully supported a 200MB daily data feed processed in under 45 seconds.

Figure 2. The gui of 60 days saas stock market analytics



Developer Dashboard
User management and platform overview

Total Users: 7
Active Users: 7 (100% of total)
Verified Users: 7 (100% of total)
Developers: 4 (1 admins)

Add New User
Create a new user account or send an invitation

User Management
Manage user accounts and permissions

User	Role	Status	Email	Last Login	Actions
HH Hilmy Haidar	Developer	Active Verified	hilmyhaidar@hintstechnology.com	19/03/2026 03:23:38	Your account
MR Muhammad Iqbal Rabani	Developer	Active Verified	muhiqbalrabani@gmail.com	18/03/2026 10:54:04	Suspend Delete
NY Nic Yansen	Admin	Active Verified	nicyansen@gmail.com	17/03/2026 07:47:10	Suspend Delete
ND Nathan Daud	Developer	Active Verified	nathandaud123@gmail.com	16/03/2026 11:39:43	Suspend Delete
ND Nathan Daud	Developer	Active Verified	nathandaud@student.ub.ac.id	15/03/2026 00:05:09	Suspend Delete
MY Mikel Yosua	User	Active Verified	mikelyosua@gmail.com	09/03/2026 05:59:57	Suspend Delete

Figure 3. Developer dashboard

Table 2. Development efficiency: vibe coding vs. estimated manual effort

Development Phase	**Baseline (Days)**	**Actual (Days)**	**Efficiency Gain (%)**
Visualize & Integrate	10	6	60%
Build (API & Logic)	35	17	52%
Execute (Deployment)	10	7	30%
Total	60	30	50%

The performance of the VIBE framework in relation to the stability–velocity paradox is evaluated through a comparison between AI-assisted development duration and a professional baseline, as presented in Table 2, derived from triangulated estimates provided by three senior developers. The results demonstrate an overall efficiency improvement of 50%, reflected in the reduction of the development cycle from 60 days to 30 days. The most substantial gain is observed in the Visualize and Integrate phases (60%), indicating the effectiveness of AI in supporting architectural formulation and early-stage system design. In contrast, the Execute phase records a comparatively lower efficiency gain (30%), which is associated with the increased requirement for human intervention to ensure system stability, validation rigor, and comprehensive hardening processes.

The transferability of the VIBE framework was established through a rigorous pattern mapping exercise, which identified that the high-concurrency and security-access protocols validated in the financial analytics platform are directly applicable to EMR modernization. By using the stock market platform as a technical proxy, the study provides a validated blueprint for health-domain infrastructures that require similar data-ingestion and interoperability standards. This mapping suggests that the 60-day roadmap utilized in this study can circumvent the risks of multiyear upgrade failures in clinical settings by treating AI as a constrained executor of pre-validated architectural patterns.

Table 3. Functional verification and performance metrics

Module	Metric/ Test Case	Observed Result	Status
Data Ingestion	Daily EOD CSV Ingestion	200MB processed in <45s	Pass
API Gateway	Rate limiting (100 req/min)	Effective throttling applied	Pass
Visualization	Rendering 5 Year Data	Chart render time: 1.2s	Pass
System Uptime	60 Days Monitoring	99.9% availability	Pass

With an uptime of 99.9% over 60 days, an average latency of 1.2 seconds, and the capacity to process 200MB of data in less than 45 seconds, this system has proven to be quite dependable, as Table 3 illustrates. This performance's validity is not merely an estimate; it is an empirical result that has been confirmed by 42 black-box tests, data triangulation from GitHub/Azure, and visual proof on the real-time dashboard. Through 70% human involvement in the validation and debugging process, the security and stability of the final findings are also guaranteed, guaranteeing that the system is not only quick but also accurate and dependable.

Table 4. Distribution of human and ai contribution per activity

Activity	Human Contribution	AI Contribution	Observation Basis
Architecture & DB Schema	90 %	10 %	Static Design
Boilerplate & Syntax	15 %	85 %	Auto-generation
Complex Business Logic	45 %	55 %	Prompt Refinement
Validation & Debugging	70 %	30 %	Unit Testing

Table 4 highlights the close relationship between humans and artificial intelligence (AI), with humans acting as a cognitive jangkar that completes 90% of architectural designs while AI acts as an efficient that completes 85% of coding tasks. The data in Table 4 is an objective analysis of 124 logs of interactions using the process tracing method. Through the Architectural Anchor mechanism, this study ensures that AI is always present in order to prevent systemic structural drift. With a human intervention of approximately 70% throughout the validation and debugging phases, it is believed that this kind of collaboration can enhance the development of a system that is aman, deterministic, and of high quality.

Discussion

The empirical evaluation confirms that the VIBE framework produces substantial improvements in development efficiency while maintaining enterprise-level system reliability. The observed reduction of the development cycle by 50%, accompanied by stable system performance indicated by 99.9% uptime and an average latency of 1.2 seconds, demonstrates that efficiency gains can coexist with architectural integrity under controlled conditions. The distribution of contributions reveals a structured collaboration pattern in which human actors dominate architectural design and validation processes, while AI primarily facilitates code generation and syntactic execution. Evidence derived from CI/CD logs and process tracing further indicates that this configuration enables consistent system coherence throughout the development lifecycle.

These findings reinforce the stability velocity paradox as a defining characteristic of AI-assisted software engineering. Hossain et al. (2025) emphasize the necessity of governance mechanisms in maintaining consistency within digital infrastructures, while Esmaeilzadeh (2024) highlights the complexity introduced by large-scale AI deployment. The results indicate that development acceleration can be aligned with structural integrity through deterministic control mechanisms. The Human–AI Cognitive Alignment model provides a conceptual framework in which human cognition functions as the primary regulator of system architecture, while AI operates within bounded execution roles. In this configuration, architectural anchoring emerges as a stabilizing mechanism that constrains non-deterministic outputs, a concern also articulated by Liu et al. (2026) in relation to hallucination in AI-generated code.

The findings demonstrate consistency with prior research while extending the current body of knowledge through empirical validation at the architectural level. Wang and Wu (2025), and Romeo and Conti (2026) report substantial productivity improvements associated with AI-assisted development, particularly in code-level tasks; however, their analyses remain concentrated at the micro-development scale. In contrast, the present study situates these efficiency gains within a broader system-level context by incorporating architectural governance as a central analytical dimension. Saravanos and Curinga (2023) identify limitations in traditional SDLC models when applied to dynamic environments, while Leong et al. (2023) highlight the absence of formal control mechanisms in hybrid development approaches. Similarly, Arshad et al. (2025) propose scalable frameworks, yet deterministic architectural control remains insufficiently articulated. The results of this study provide empirical evidence that structured governance mechanisms, operationalized through architectural anchoring, can sustain system integrity alongside efficiency improvements within real-world AI-assisted development environments

An important observation emerges in the comparatively lower efficiency gain within the Execute phase. This phase involves deployment, validation, and system hardening processes that require strict determinism and precision. Huang et al. (2025) underline the critical role of human intervention in controlling AI behavior within complex systems, particularly during operational stages that demand accountability and traceability. The findings indicate that AI contribution decreases in contexts requiring high levels of reliability and compliance with system constraints. This pattern suggests that efficiency improvements are not uniformly distributed across development phases and that human oversight remains essential in ensuring system stability during final-stage operations.

The study contributes to software engineering research by formalizing architectural anchoring as a mechanism for maintaining structural integrity in AI-assisted development environments. Medvidović (2025) emphasizes the importance of addressing systemic complexity in contemporary software engineering, which aligns with the proposed Human–AI Cognitive Alignment model. This model introduces a structured interaction paradigm in which human expertise governs architectural decisions while AI operates within constrained execution boundaries. In addition, the integration of quantitative performance analysis with qualitative process tracing provides a comprehensive methodological approach that links micro-level efficiency with macro-level governance, an aspect that remains underexplored in prior research.

The findings also present practical implications for both practitioners and policymakers. Kang and Park (2025) highlight the importance of adaptive system architectures in AI-driven environments, while Pesce and Cheungpasitporn (2025) emphasize the necessity of human oversight in sensitive infrastructures such as EMR. The VIBE framework offers a structured development model that enables efficient integration of AI capabilities within controlled architectural boundaries. This model supports the development of governance strategies that prioritize human control in AI-assisted workflows, particularly in sectors requiring high levels of data integrity and regulatory compliance. The use of financial analytics as a high-concurrency proxy further demonstrates the applicability of the framework to other mission-critical domains.

The study is subject to certain limitations. The development baseline relies on expert estimation, which may introduce bias related to developer experience, as noted by Meske et al. (2025). In addition, the empirical scope is confined to a single case context, which may influence the generalizability of the findings across different domains and development environments. Future research may expand the empirical scope by incorporating multiple case studies and varying levels of developer expertise to enhance external validity. Longitudinal investigations are also required to examine the accumulation of technical debt in AI-assisted systems over extended periods. Furthermore, the exploration of multi-agent AI orchestration, as discussed by Huang et al. (2025), represents a promising direction for advancing hybrid SDLC models while maintaining strict architectural control

CONCLUSION

The results indicate that the VIBE framework facilitates significant gains in development efficiency while ensuring the preservation of architectural integrity in AI assisted software development contexts. Empirical evidence shows that the integration of human-driven architectural governance with AI-supported execution reduces development duration by up to 50%, without degrading system reliability, as reflected in consistent uptime and stable performance indicators. This study advances the Human AI Cognitive Alignment model, positioning AI as a bounded executor within clearly defined architectural constraints, and establishes architectural anchoring as a key mechanism for maintaining system coherence. These findings contribute to bridging the relationship between efficiency at the operational level and governance at the structural level, offering both theoretical enrichment and practical applicability. The proposed framework provides a validated approach for the development of mission-critical systems, including EMR, while highlighting the importance of further empirical validation across varied development environments and longer-term system lifecycles to enhance generalizability and robustness

REFERENCES

Al-Ghuraybi, H. A., AlZain, M. A., & Soh, B. (2023). Ensuring authentication in Medical Cyber-Physical Systems: A comprehensive literature review of blockchain technology integration with machine learning. *Multimedia Tools and Applications*, *83*(12), 35673–35707. https://doi.org/10.1007/s11042-023-17065-3

Arshad, N., Butt, T. A., & Iqbal, M. (2025). A Comprehensive Framework for Intelligent, Scalable, and Performance-Optimized Software Development. *IEEE Access*, *13*, 74062–74077. https://doi.org/10.1109/ACCESS.2025.3564139

Esmaeilzadeh, P. (2024). Challenges and strategies for wide-scale artificial intelligence (AI) deployment in healthcare practices: A perspective for healthcare organizations. *Artificial Intelligence in Medicine*, *151*, 102861. https://doi.org/10.1016/j.artmed.2024.102861

Gabarron, E., Larbi, D., Rivera-Romero, O., & Denecke, K. (2024). Human Factors in AI-Driven Digital Solutions for Increasing Physical Activity: Scoping Review. *JMIR Human Factors*, *11*, e55964. https://doi.org/10.2196/55964

Hossain, Md. K., Sutanto, J., Handayani, P. W., Haryanto, A. A., Bhowmik, J., & Frings-

Hessami, V. (2025). An exploratory study of electronic medical record implementation and recordkeeping culture: the case of hospitals in Indonesia. *BMC Health Services Research*, *25*(1), 249. https://doi.org/10.1186/s12913-025-12399-0

Hovde, M. R., & Threatt, B. (2025). The Nature and Indispensable Roles of Technical Communication in Agile Development Environments: Following Typical Processes and Adapting to Address Challenges. *IEEE Transactions on Professional Communication*, *68*(3), 302–321. https://doi.org/10.1109/TPC.2025.3585658

Huang, Z., Sheng, Z., Wan, Z., Qu, Y., Luo, Y., Wang, B., ... & Chen, S. (2025). Sky-Drive: a Distributed Multiagent Simulation Platform for Human-AI Collaborative and Socially Aware Future Transportation. *Journal of Intelligent and Connected Vehicles*, *8*(4), 9210070-1. https://doi.org/10.26599/JICV.2026.9210070

Kang, M., & Park, D. (2025). Flexible Edge-AI Software Execution Architecture Based on Cloud-Connected Incremental Learning. *IEEE Access*, *13*, 120772–120784. https://doi.org/10.1109/ACCESS.2025.3586940

Khan, A. N., Mehmood, K., & Soomro, M. A. (2024). Knowledge Management-Based Artificial Intelligence (AI) Adoption in Construction SMEs: The Moderating Role of Knowledge Integration. *IEEE Transactions on Engineering Management*, *71*, 10874–10884. https://doi.org/10.1109/TEM.2024.3403981

Kopuz, E., & Kartal, G. (2025). Collaborative Human–AI Research Practices: Identifying Critical Touchpoints for Human Intervention in Educational Research. *IEEE Transactions on Learning Technologies*, *18*, 732–740. https://doi.org/10.1109/TLT.2025.3587488

Leong, J., May Yee, K., Baitsegi, O., Palanisamy, L., & Ramasamy, R. K. (2023). Hybrid Project Management between Traditional Software Development Lifecycle and Agile Based Product Development for Future Sustainability. *Sustainability*, *15*(2), 1121. https://doi.org/10.3390/su15021121

Liu, F., Liu, Y., Shi, L., Yang, Z., Zhang, L., Lian, X., Li, Z., & Ma, Y. (2026). Beyond Functional Correctness: Exploring Hallucinations in LLM-Generated Code. *IEEE Transactions on Software Engineering*, *52*(3), 1037–1055. https://doi.org/10.1109/TSE.2026.3657432

Medvidović, N. (2025). Software Engineering Research Trends 1994–2024: Stepping Beyond the Lamppost. *IEEE Transactions on Software Engineering*, *51*(3), 685–688. https://doi.org/10.1109/TSE.2025.3530898

Meske, C., Hermanns, T., Von der Weiden, E., Loser, K.-U., & Berger, T. (2025). Vibe Coding as a Reconfiguration of Intent Mediation in Software Development: Definition, Implications, and Research Agenda. *IEEE Access*, *13*, 213242–213259. https://doi.org/10.1109/ACCESS.2025.3645466

Pesce, F., & Cheungpasitporn, W. (2025). Vibe Coding in nephrology education: clinician-led, AI-assisted development of open-source interactive learning tools. *Renal Failure*, *47*(1). https://doi.org/10.1080/0886022X.2025.2581933

Rahaman, M. S., Tisha, S. N., Song, E., & Cerny, T. (2023). Access Control Design Practice and Solutions in Cloud-Native Architecture: A Systematic Mapping Study. *Sensors*, *23*(7), 3413. https://doi.org/10.3390/s23073413

Romeo, G., & Conti, D. (2026). Exploring automation bias in human–AI collaboration: a review and implications for explainable AI. *AI & SOCIETY*, *41*(1), 259–278. https://doi.org/10.1007/s00146-025-02422-7

Saravanos, A., & Curinga, M. X. (2023). Simulating the Software Development Lifecycle: The Waterfall Model. *Applied System Innovation*, *6*(6), 108. https://doi.org/10.3390/asi6060108

Wang, Y., & Wu, Y. (2025). Digital Economy, Rural E-Commerce Development, and Farmers'

Employment Quality. *Sustainability*, *17*(7), 2949. https://doi.org/10.3390/su17072949

Yang, J., Guo, J., & Gong, L. (2026). Against Techno-Feudalism in Academia: Reclaiming Scholarly Autonomy in the Age of AI. *IEEE Transactions on Learning Technologies*, *19*, 249–254. https://doi.org/10.1109/TLT.2026.3673243

Yang, Y., Wang, T., & Xiang, W. (2025). A Distributed Neural Hybrid System Learning Framework in Modeling Complex Dynamical Systems. *IEEE Transactions on Neural Networks and Learning Systems*, *36*(5), 9463–9473. https://doi.org/10.1109/TNNLS.2024.3417330

Yitagesu, S., Xing, Z., Zhang, X., Feng, Z., Bi, T., Han, L., & Li, X. (2026). Systematic Literature Review on Software Security Vulnerability Information Extraction. *ACM Transactions on Software Engineering and Methodology*, *35*(4), 1–52. https://doi.org/10.1145/3745026