DevOps Techniques for Artificial Intelligence Development and Training: A Human-Centered Approach to AI-Driven Operations

Kateryna Shulakova[1,2], Liliia Bodnar[3], Mykola Bodnar[4], Roman Tsarov[1], Olha Yavorska[1], Vladyslav Kumysh[1] and Oleksandra Ordanovska[3]

[1]*State University of Intelligent Technologies and Telecommunications, Kuznechna Str. 1, Odesa, Ukraine*

[2]*Anhalt University of Applied Sciences, Bernburger Str. 57, Köthen, Germany*

[3]*South Ukrainian National Pedagogical University, Staroportofrankyvska Str. 26, 65020 Odesa, Ukraine*

[4] *LLC B&B Solutions, Dukivska Str. 5 65023 Odesa, Ukraine*

katejojo29@gmail.com, bodnarl79@pdpu.edu.ua, rcarev@gmail.com, yavorskayao7@gmail.com, vlad.kumish@gmail.com, aleksordanovskaya@pdpu.edu.ua

Keywords: DevOps, AIOps, MLOps, DevSecOps, Artificial Intelligence, Automation Paradox, Human-Centered AI, Intelligent Monitoring, Cloud-Native Infrastructure, Operational Governance, CI/CD, Infrastructure as Code, Kubernetes, AI Operations, Predictive Analytics.

Abstract: The rapid integration of artificial intelligence technologies into software engineering and cloud-native infrastructures has significantly transformed modern operational practices and automation strategies. Traditional DevOps methodologies based on continuous integration and continuous deployment, infrastructure automation, and container orchestration remain essential for scalable software delivery; however, the increasing complexity of distributed AI-oriented systems creates new operational, organizational, and security-related challenges. The emergence of intelligent operational paradigms such as AIOps, MLOps, and DevSecOps demonstrates the growing need for adaptive monitoring, predictive analytics, automated decision support, and secure lifecycle management within AI-driven infrastructures. This paper investigates the application of DevOps techniques in artificial intelligence development and training environments from a human-centered operational perspective. Particular attention is devoted to the automation paradox, where increasing automation levels may simultaneously improve operational efficiency while generating hidden technical complexity, infrastructure dependency, and reduced engineering visibility. The study analyzes the role of intelligent monitoring, anomaly detection, predictive scaling, and automated operational analytics in modern cloud ecosystems while emphasizing the importance of human expertise in supervision, validation, and governance processes. The paper proposes an integrated conceptual framework combining DevOps, AIOps, MLOps, and DevSecOps principles for sustainable AI operational management. Additionally, a conceptual automation-risk model is introduced to describe the relationship between automation intensity, infrastructure complexity, data uncertainty, and engineering expertise. The results demonstrate that artificial intelligence should function as an augmentation mechanism for engineering decision-making rather than a replacement for human operational control. The proposed approach contributes to the development of secure, scalable, and sustainable AI-driven infrastructures capable of balancing automation efficiency with governance, transparency, and operational reliability.

1 INTRODUCTION

The rapid development of artificial intelligence (AI) technologies has significantly transformed modern software engineering, cloud computing, and infrastructure management processes. AI-driven systems are increasingly integrated into distributed cloud-native environments, automated operational workflows, and large-scale data-processing infrastructures. Simultaneously, the growing complexity of modern software ecosystems requires new approaches to automation, scalability, operational monitoring, and infrastructure governance [1]-[3].

The DevOps paradigm, based on continuous integration and continuous deployment (CI/CD), infrastructure automation, and collaborative operational practices, has become one of the dominant methodologies for accelerating software

delivery and improving infrastructure reliability [1]-[3]. The emergence of containerization and orchestration platforms such as Docker and Kubernetes further enabled scalable deployment of distributed services and microservice-based architectures [4], [5]. In parallel, Infrastructure as Code (IaC) approaches improved reproducibility and operational consistency in cloud-native infrastructures [21].

However, despite the significant advantages of classical DevOps automation, modern AI-oriented infrastructures increasingly exceed the capabilities of traditional rule-based operational management. Large-scale AI systems require continuous model retraining, dynamic resource allocation, intelligent anomaly detection, and predictive operational analytics [6], [12], [13]. Existing studies demonstrate that machine learning systems introduce additional operational complexity associated with hidden technical debt, model drift, dependency management, and continuous lifecycle maintenance [6], [13].

To address these challenges, new operational paradigms such as Machine Learning Operations (MLOps), Artificial Intelligence for IT Operations (AIOps), and DevSecOps have emerged. MLOps extends DevOps principles toward machine learning lifecycle management, including dataset versioning, model training, deployment, and retraining pipelines [7], [12], [16]. AIOps integrates machine learning algorithms into operational monitoring systems for anomaly detection, predictive analytics, and intelligent event correlation in complex infrastructures [9], [17]. DevSecOps further incorporates security validation and compliance mechanisms into CI/CD workflows to improve operational resilience and secure software delivery [11].

Despite the rapid adoption of AI-enhanced operational frameworks, the increasing dependence on intelligent automation creates new organizational and technical risks. One of the most important challenges is the so-called automation paradox, where higher levels of automation may simultaneously reduce manual workload while increasing hidden infrastructural complexity and dependence on expert supervision. In practice, fully autonomous operational systems remain unattainable because AI technologies still require human validation, governance, and strategic decision-making [6], [9].

Furthermore, excessive reliance on AI-generated recommendations may reduce engineering visibility and operational awareness in critical infrastructures. Intelligent systems can improve operational efficiency and scalability; however, they may also generate explainability limitations, false-positive recommendations, infrastructure fragility, and operational overconfidence [9], [20]. Consequently, the role of the engineer shifts from direct operational execution toward supervision, governance, anomaly interpretation, and strategic control of intelligent operational ecosystems.

Security considerations also become increasingly important in AI-oriented infrastructures. AI systems introduce additional attack surfaces related to training datasets, model artifacts, operational telemetry, and automated decision-making mechanisms. International standards and governance frameworks emphasize the importance of transparency, accountability, risk management, and continuous monitoring across the entire AI lifecycle [14], [15]. Therefore, sustainable AI operations require not only intelligent automation technologies but also governance-oriented operational models capable of balancing automation efficiency, security, transparency, and human-centered supervision.

This paper investigates the application of DevOps techniques in artificial intelligence development and training environments from a human-centered operational perspective. The study focuses on the integration of DevOps, AIOps, MLOps, and DevSecOps methodologies in cloud-native infrastructures while addressing the limitations of intelligent automation and the operational risks associated with excessive dependence on AI-driven systems.

The main contributions of this paper are as follows:

1) Development of a conceptual framework integrating DevOps, AIOps, MLOps, and DevSecOps methodologies for AI operational management.
2) Formalization of the automation paradox in AI-driven infrastructures.
3) Analysis of operational risks associated with intelligent automation and hidden technical complexity.
4) Proposal of a human-centered governance model for sustainable AI operations.
5) Introduction of an analytical automation-risk formulation for evaluating operational sustainability in AI-enhanced infrastructures.

2 RELATED WORK

2.1 DevOps Foundations and Continuous Delivery

The DevOps paradigm emerged as a response to the need for faster, more reliable, and continuously evolving software delivery processes. Continuous integration and continuous delivery (CI/CD) form the

methodological core of DevOps, enabling automated testing, integration, and deployment pipelines across complex software systems [1]. These principles emphasize collaboration between development and operations teams and establish continuous feedback loops for improving software quality and deployment efficiency [2].

From an architectural perspective, DevOps represents a shift from isolated development cycles toward integrated lifecycle management, where infrastructure, deployment, and monitoring are treated as continuous processes rather than discrete stages [3]. This transformation has become a foundational concept for modern cloud-native systems.

2.2 Cloud-Native Infrastructure and Automation

The evolution of containerization technologies has significantly accelerated the adoption of DevOps practices. Docker introduced lightweight virtualization for application packaging and deployment, enabling portability across heterogeneous environments [4]. Kubernetes further extended these capabilities by providing orchestration, scaling, and automated resource management for distributed microservice architectures [5].

Infrastructure as Code (IaC) represents another critical advancement in cloud automation, allowing infrastructure provisioning and configuration to be managed through version-controlled code. Recent studies demonstrate that IaC improves reproducibility, scalability, and operational consistency in cloud-based environments while reducing manual configuration errors [6]. However, increasing automation complexity also introduces additional operational overhead and dependency on tooling ecosystems.

2.3 Machine Learning Systems and Operational Complexity

Unlike traditional software systems, machine learning applications introduce additional lifecycle complexity due to their dependence on data, model retraining, and continuous evaluation. Sculley et al. identified that machine learning systems accumulate significant “hidden technical debt” caused by data dependencies, pipeline fragility, and evolving model behavior [7].

To address these challenges, Machine Learning Operations (MLOps) has emerged as an extension of DevOps principles to machine learning lifecycle management. MLOps integrates dataset versioning, model training, deployment, monitoring, and retraining into unified operational pipelines [8]. Empirical studies further highlight that engineering machine learning systems requires specialized workflows that differ significantly from traditional software engineering practices [9].

Frameworks such as MLflow provide infrastructure for experiment tracking, model versioning, and lifecycle reproducibility, enabling more structured management of AI systems in production environments [10]. Nevertheless, continuous delivery of machine learning models remains challenging due to data drift, model instability, and evaluation uncertainty.

2.4 Continuous Delivery Challenges in Modern Systems

Despite its advantages, continuous delivery introduces operational and organizational challenges in large-scale systems. Research indicates that frequent deployments increase system complexity, require stronger monitoring mechanisms, and introduce new risks related to system stability and rollback management [11]. These challenges become more pronounced in AI-driven environments where models evolve continuously and depend on dynamic datasets.

Industrial implementations of MLOps pipelines demonstrate that automation must be carefully balanced with validation mechanisms and human oversight to ensure system reliability and consistency [12].

2.5 AIOps and Intelligent Operations

Artificial Intelligence for IT Operations (AIOps) represents a shift toward applying machine learning techniques for operational monitoring, anomaly detection, and predictive analytics in complex IT environments. AIOps systems aggregate and analyze large volumes of telemetry data to identify system anomalies and support decision-making processes in real time [13].

Cloud-native automation frameworks further extend AIOps capabilities by integrating intelligent monitoring and adaptive scaling mechanisms across distributed infrastructures [14]. However, the effectiveness of AIOps systems strongly depends on data quality, system observability, and model interpretability, which remain open research challenges.

2.6 DevSecOps and Security in AI Systems

Security considerations have become increasingly important in modern DevOps pipelines. DevSecOps integrates security validation, vulnerability scanning, and compliance checks directly into CI/CD workflows, ensuring that security is addressed throughout the entire software lifecycle rather than as a post-deployment activity [15].

In AI-driven systems, security concerns are amplified due to additional attack surfaces such as training data manipulation, model poisoning, and adversarial inputs. International standards such as ISO/IEC 5338 and the NIST AI Risk Management Framework emphasize lifecycle-wide governance, transparency, and accountability in AI systems [16], [17].

2.7 AI Systems and Theoretical Foundations

Artificial intelligence systems operate on foundational principles of learning, reasoning, and adaptation, which are extensively formalized in classical AI literature [18]. However, the integration of AI into operational environments introduces new challenges that extend beyond theoretical AI models into real-world engineering constraints.

2.8 Practical Implementations and Industrial Experience

Industrial case studies demonstrate that successful DevOps adoption requires not only technical tooling but also organizational maturity and structured engineering practices [18]. Large-scale DevOps implementations highlight the importance of standardized pipelines, automated testing, and continuous monitoring for maintaining system stability in production environments [19].

Further research shows that infrastructure cost optimization and resource efficiency are critical factors in container-based cloud deployments, particularly in large-scale distributed systems [21].

2.9 Research Gap

Although extensive research exists in DevOps, cloud-native systems, MLOps, AIOps, and DevSecOps independently, there is still a lack of unified frameworks that integrate these paradigms under a human-centered operational model. In particular, the interaction between automation complexity, AI-driven decision-making, and human supervision remains insufficiently formalized.

This gap motivates the need for a unified approach that addresses the automation paradox in AI-driven DevOps infrastructures and balances operational efficiency with governance, transparency, and human control.

Although the automation paradox has been theoretically described in classical human factors research [22], its formalization within the specific context of AI-driven DevOps and cloud-native infrastructures - integrating MLOps, AIOps, and DevSecOps simultaneously - has not been systematically addressed in the existing literature. Furthermore, prior work has not proposed a governance-oriented risk model that explicitly incorporates data uncertainty and governance maturity as operational variables.

3 AUTOMATION PARADOX THEORY

The increasing integration of artificial intelligence into DevOps and cloud-native operational environments has intensified the role of automation in modern infrastructure management. Automated deployment pipelines, intelligent monitoring systems, predictive scaling, and AI-enhanced operational analytics significantly improve deployment speed, scalability, and infrastructure adaptability [1], [2], [13]. However, despite these advantages, the growing dependence on intelligent automation introduces a critical contradiction commonly referred to as the automation paradox.

The concept of the automation paradox is not new to engineering science. Bainbridge [22] identified the fundamental "ironies of automation" as early as 1983, demonstrating that automated systems paradoxically increase the cognitive demands placed on human operators rather than eliminating them. In modern AI-driven DevOps infrastructures, this phenomenon is significantly amplified. The automation paradox describes a situation in which increasing levels of automation simultaneously reduce routine manual effort while increasing hidden operational complexity and dependence on expert supervision. Instead of eliminating human involvement, automation transforms the role of engineers from direct operational execution toward supervision, governance, anomaly interpretation, and strategic decision-making [7], [13].

In AI-driven infrastructures, this phenomenon becomes particularly significant due to the adaptive and probabilistic nature of machine learning systems.

Unlike deterministic software automation, AI systems continuously evolve through changing datasets, retraining processes, and dynamic operational environments [8], [9]. As a result, fully autonomous operation remains unattainable because intelligent systems require continuous validation, monitoring, and contextual interpretation by human experts.

Existing studies on machine learning engineering demonstrate that AI systems accumulate hidden technical debt caused by unstable dependencies, evolving datasets, configuration drift, and pipeline fragility [7]. These factors increase operational uncertainty and reduce infrastructure transparency, particularly in highly distributed cloud-native ecosystems. Consequently, organizations adopting AI-enhanced DevOps practices frequently encounter situations where automation simplifies low-level operational tasks while simultaneously increasing systemic fragility at the architectural level.

The paradox is further amplified in AIOps environments, where machine learning algorithms process large volumes of telemetry data and generate operational recommendations in real time [13], [14]. Although intelligent monitoring systems improve anomaly detection and predictive analytics capabilities, excessive dependence on AI-generated recommendations may reduce engineering visibility and operational awareness. Engineers may increasingly rely on automated decision support mechanisms without fully understanding the underlying operational context or model limitations.

From a theoretical perspective, the automation paradox can be represented as the nonlinear relationship between automation intensity and operational complexity. At low and moderate levels, automation reduces repetitive workload and improves infrastructure efficiency. However, beyond a certain threshold, additional automation introduces hidden dependencies, explainability limitations, integration complexity, and increased governance requirements.

This relationship may be conceptually represented as:

$$R = \alpha A + \beta C + \gamma D - \delta H$$

where: R - operational automation risk; A - automation intensity; C - infrastructure complexity; D - data uncertainty and model drift; H - level of human expertise and operational supervision; $\alpha,\beta,\gamma,\delta$ - weighting coefficients describing infrastructure sensitivity.

The structure of this formulation is consistent with established risk-modeling approaches in systems engineering, where operational risk is expressed as a product of likelihood factors modulated by control mechanisms [23]. The weighting coefficients α, β, γ, δ are infrastructure-specific and should be empirically calibrated for each deployment context. In the absence of empirical data, this model serves as a qualitative decision-support tool rather than a deterministic predictive instrument - a limitation explicitly acknowledged by the authors. Quantitative calibration of these coefficients remains a direction for future empirical research.

The model illustrates that operational risk increases proportionally with automation intensity, infrastructural complexity, and uncertainty in AI-driven decision-making, while human expertise functions as a stabilizing factor that reduces systemic fragility.

Another important characteristic of the automation paradox is the asymmetry between operational efficiency and explainability. AI systems may provide highly efficient recommendations while remaining partially opaque to operators. This creates a governance challenge in critical infrastructures where accountability and interpretability remain essential operational requirements [15]-[17].

Furthermore, intelligent automation may contribute to gradual degradation of engineering intuition. When engineers become excessively dependent on AI-generated operational recommendations, their ability to independently diagnose failures and interpret complex infrastructure behavior may decline over time. This issue is especially critical in large-scale cloud-native systems where cascading failures require rapid contextual reasoning beyond predefined automation scenarios [5], [18].

Therefore, the automation paradox demonstrates that sustainable AI operations cannot rely solely on increasing automation levels. Instead, effective AI-driven DevOps infrastructures require human-centered governance models that combine intelligent automation with transparency, explainability, operational visibility, and continuous expert supervision.

4 HUMAN-CENTERED AIOPS FRAMEWORK

The increasing complexity of AI-driven infrastructures requires operational models capable of combining intelligent automation with continuous human supervision. Existing AIOps implementations primarily focus on anomaly detection, predictive analytics, and automated operational responses;

however, excessive automation may reduce infrastructure transparency and introduce additional governance risks [13], [14]. Therefore, this study proposes a human-centered AIOps framework designed to balance operational efficiency with explainability, accountability, and engineering control.

The proposed framework integrates DevOps, MLOps, AIOps, and DevSecOps principles into a unified operational ecosystem. The framework consists of five interconnected layers:

1) Infrastructure layer - cloud-native environments, container orchestration platforms, and Infrastructure as Code mechanisms [4]-[6].
2) Operational data layer - telemetry collection, logs, traces, metrics, and event aggregation.
3) Intelligence Layer - machine learning models responsible for anomaly detection, predictive analytics, and operational recommendations [8], [13].
4) Governance layer - policy validation, explainability mechanisms, security controls, and compliance management [15]-[17].
5) Human supervision layer - expert validation, strategic decision-making, incident interpretation, and operational governance.

The key principle of the framework is that AI-generated operational actions should remain bounded by human-centered validation policies. Automated remediation may be acceptable for low-risk operational tasks, while critical infrastructural changes require human approval before execution.

The proposed framework also emphasizes operational transparency. All AI-generated recommendations must remain observable, explainable, and traceable within the operational lifecycle. This approach reduces the probability of cascading failures caused by opaque decision-making mechanisms and improves infrastructure resilience in highly dynamic environments.

4.1 Risk Model

Operational sustainability in AI-driven infrastructures depends on the interaction between automation intensity, infrastructure complexity, data quality, and human supervision. To formalize this relationship, the proposed study introduces a conceptual operational risk model.

The overall operational risk may be expressed as:

$$R = \frac{A(C + D + S)}{H + G}$$

where: S - security exposure; G - governance maturity.

The model demonstrates that operational risk increases proportionally with automation intensity, infrastructure complexity, uncertainty, and security exposure. Simultaneously, risk decreases as human expertise and governance maturity increase.

The relationship described by the model is nonlinear. At moderate automation levels, operational efficiency improves due to reduced manual workload and faster response times. However, excessive automation may generate hidden dependencies, reduced explainability, and operational fragility. This effect becomes especially visible in highly distributed infrastructures characterized by dynamic scaling, autonomous remediation, and adaptive machine learning models.

The proposed formulation does not represent a deterministic mathematical prediction model; instead, it functions as a conceptual analytical framework for evaluating operational sustainability in intelligent infrastructures.

4.2 Comparative Analysis

To evaluate the role of intelligent operational paradigms in modern infrastructures, a comparative analysis of traditional DevOps automation and AI-enhanced operational approaches was performed in Table 1. The characteristics attributed to each paradigm are derived from the following sources: Traditional DevOps [1]-[3]; MLOps [8], [12]; AIOps [13], [14]. The Human-Centered AIOps column reflects the proposed framework introduced in this study.

The analysis demonstrates that traditional DevOps environments provide high operational transparency but limited adaptive capabilities. In contrast, AIOps systems improve scalability and predictive analytics but may reduce explainability and increase dependence on automated recommendations.

The proposed human-centered AIOps model attempts to balance these trade-offs by preserving operational visibility and governance while still leveraging intelligent automation capabilities.

Table 1: Comparative analysis of operational paradigms.

Characteristic	Traditional DevOps	MLOps	AIOps	Human-Centered AIOps
Primary Focus	Software delivery	ML lifecycle	Intelligent operations	Governance-oriented automation
Automation Type	Rule-based	Pipeline-based	Predictive/adaptive	Supervised intelligent automation
Human Involvement	Moderate	High	Reduced	Continuous supervision
Explainability	High	Medium	Low-Medium	High
Operational Risk	Moderate	High	High	Controlled
Security Integration	DevSecOps	Secure ML pipelines	Monitoring-oriented	Governance-integrated
Decision Transparency	High	Medium	Low	High

5 DISCUSSION

The results of this study demonstrate that intelligent automation significantly changes the operational dynamics of cloud-native infrastructures. While AI-enhanced operational systems improve scalability, anomaly detection, and predictive analytics capabilities, they also introduce additional complexity associated with governance, explainability, and dependency management.

One of the most important findings is that increasing automation does not eliminate the need for human expertise. Instead, the role of engineers evolves from direct operational execution toward strategic supervision, validation, and governance. This transformation aligns with the automation paradox discussed earlier, where automation simultaneously improves efficiency and increases hidden infrastructural complexity.

The study further indicates that operational sustainability depends not only on the quality of AI models but also on governance maturity, transparency mechanisms, and organizational readiness. Infrastructures lacking operational visibility and explainability may become vulnerable to cascading failures, model drift, and incorrect automated decisions.

Security also emerges as a critical dimension of AI-enhanced infrastructures. AI systems introduce additional attack surfaces related to operational telemetry, model artifacts, and automated remediation mechanisms. Therefore, secure operational governance must become an integral component of intelligent infrastructure management.

Another important implication concerns engineering education and operational culture. Excessive dependence on AI-generated operational recommendations may reduce engineering intuition and diagnostic reasoning capabilities over time. Consequently, sustainable AIOps adoption requires balancing automation efficiency with continuous human expertise development.

The proposed framework therefore supports a human-centered operational philosophy in which artificial intelligence functions as an augmentation mechanism rather than a replacement for engineering decision-making.

This study presents a conceptual framework and does not provide empirical validation in live production environments. The proposed risk model relies on qualitative weighting rather than statistically derived coefficients. The comparative analysis in Table 1 is based on a literature synthesis rather than controlled experimental conditions.

6 CONCLUSIONS

This paper investigated the application of DevOps techniques in artificial intelligence development and training environments from a human-centered operational perspective. The study analyzed the convergence of DevOps, MLOps, AIOps, and DevSecOps methodologies in modern cloud-native infrastructures and identified the operational challenges associated with excessive dependence on intelligent automation.

Particular attention was devoted to the automation paradox, which describes the contradiction between increasing automation levels and the growing need for expert supervision and governance. The results demonstrate that AI-driven operational systems improve scalability, predictive analytics, and infrastructure adaptability; however, they simultaneously introduce hidden technical complexity, explainability limitations, and governance-related risks.

To address these challenges, the paper proposed a human-centered AIOps framework integrating intelligent monitoring, governance-oriented automation, security mechanisms, and continuous

expert supervision. Additionally, a conceptual operational risk model was introduced to formalize the relationship between automation intensity, infrastructure complexity, data uncertainty, and human expertise.

The findings indicate that sustainable AI-driven infrastructures cannot rely solely on autonomous operational mechanisms. Instead, successful intelligent infrastructures require balanced integration of automation, transparency, governance, and human operational control.

Future research may focus on empirical validation of the proposed framework in real-world cloud-native infrastructures, quantitative evaluation of automation-risk relationships, and development of explainable operational AI models for critical distributed systems.

ACKNOWLEDGMENTS

We acknowledge support by the German Research Foundation (Deutsche Forschungsgemeinschaft, DFG) and the Open Access Publishing Fund of Anhalt University of Applied Sciences.

REFERENCES

[1] J. Humble and D. Farley, Continuous Delivery. Boston, MA, USA: Addison-Wesley, 2011.

[2] G. Kim, J. Humble, P. Debois, and J. Willis, The DevOps Handbook. Portland, OR, USA: IT Revolution, 2021.

[3] L. Bass, I. Weber, and L. Zhu, DevOps: A Software Architect's Perspective. Boston, MA, USA: Addison-Wesley, 2015.

[4] J. Turnbull, The Docker Book. James Turnbull, 2019.

[5] B. Burns, J. Beda, and K. Hightower, Kubernetes: Up and Running. Sebastopol, CA, USA: O'Reilly Media, 2022.

[6] L. Bodnar, M. Bodnar, K. Shulakova, O. Vasylenko, E. Siemens, R. Tsarov, O. Yavorska, and O. Tyurikova, "Advanced techniques for IaC: Enhancing automation and optimization in cloud-based infrastructure management," Proc. Int. Conf. Applied Innovation in IT, vol. 12, no. 2, pp. 19-25, 2024, doi: 10.25673/118105.2.

[7] D. Sculley et al., "Hidden technical debt in machine learning systems," in Proc. Adv. Neural Inf. Process. Syst. (NIPS), 2015.

[8] M. Shahin et al., "Machine learning operations (MLOps): Overview, definition, and architecture," IEEE Access, vol. 10, pp. 31841-31865, 2022, doi: 10.1109/ACCESS.2022.3157181.

[9] S. Amershi et al., "Software engineering for machine learning: A case study," in Proc. IEEE/ACM 41st Int. Conf. Softw. Eng. (ICSE), 2019, pp. 291-300.

[10] M. Zaharia et al., "Accelerating the machine learning lifecycle with MLflow," IEEE Data Eng. Bull., vol. 41, no. 4, pp. 39-45, 2018.

[11] L. Chen, "Continuous delivery: Huge benefits, but challenges too," IEEE Software, vol. 32, no. 2, pp. 50-54, 2015, doi: 10.1109/MS.2015.27.

[12] C. Huyen, Designing Machine Learning Systems. Sebastopol, CA, USA: O'Reilly Media, 2022.

[13] I. Rangwala et al., "AIOps: Real-world challenges and research innovations," IEEE Internet Computing, vol. 25, no. 3, pp. 81-85, 2021.

[14] G. Deng et al., "AIOps: Real-world operational AI challenges," IEEE Transactions on Services Computing, vol. 15, no. 4, pp. 2186-2199, 2022, doi: 10.1109/TSC.2020.3038824.

[15] A. Rahman et al., "DevSecOps: A multivocal literature review," Information and Software Technology, vol. 121, Art. no. 106200, 2020, doi: 10.1016/j.infsof.2020.106200.

[16] ISO/IEC 5338:2023, AI System Lifecycle Processes. Geneva, Switzerland: International Organization for Standardization, 2023.

[17] NIST, Artificial Intelligence Risk Management Framework (AI RMF 1.0). Gaithersburg, MD, USA: National Institute of Standards and Technology, 2023.

[18] S. Russell and P. Norvig, Artificial Intelligence: A Modern Approach. Hoboken, NJ, USA: Pearson, 2021.

[19] L. Bodnar, M. Bodnar, K. Shulakova, O. Vasylenko, R. Tsarov, and E. Siemens, "Practical experience in DevOps implementation," Proc. Int. Conf. Applied Innovation in IT, vol. 12, no. 1, pp. 33-39, 2024, doi: 10.25673/115639.

[20] L. Bodnar, M. Bodnar, K. Shulakova, O. Vasylenko, E. Siemens, and O. Tsyra, "A comprehensive integration of practical strategies in DevOps," Lecture Notes in Networks and Systems. Cham, Switzerland: Springer, 2025, pp. 336-359.

[21] M. Villamizar et al., "Infrastructure cost comparison of container-based cloud deployment," Future Generation Computer Systems, vol. 79, pp. 619-633, 2018, doi: 10.1016/j.future.2017.08.028.

[22] L. Bainbridge, "Ironies of automation," Automatica, vol. 19, no. 6, pp. 775-779, 1983, doi: 10.1016/0005-1098(83)90046-8.

[23] IEC 31010:2019, Risk Management - Risk Assessment Techniques. Geneva, Switzerland: International Electrotechnical Commission, 2019.