pISSN:2349-5707 Volume:13, Issue 04, April-2026 eISSN:2349-5715

Role of Upfront Protection Evaluation in Identifying System Deficiencies during Integrated Development Operations

Aram Hakobyan
Yerevan State University, Armenia

ABSTRACT: Modern integrated development operations increasingly rely on automated pipelines, distributed system architectures, and continuous deployment mechanisms. While these advancements improve efficiency and scalability, they also amplify system complexity and introduce hidden vulnerabilities that often remain undetected until late stages of the development lifecycle. This research investigates the role of upfront protection evaluation as a proactive mechanism for identifying system deficiencies during integrated development operations. The study conceptualizes upfront protection evaluation as a structured pre-implementation assessment framework that identifies risks, inefficiencies, and structural weaknesses before system execution begins.

Drawing from interdisciplinary domains such as satellite systems engineering, autonomous operations, and large-scale monitoring infrastructures, the research constructs a theoretical model that aligns early validation principles with software development workflows. Studies on satellite mission planning and operational systems (Gathmann & Raslavicius, 1990; Sweeting, 2018) demonstrate that early system evaluation improves reliability and reduces operational failure rates. Similarly, autonomous system frameworks (Anderson et al., 2009) emphasize predictive evaluation as a critical factor in ensuring system robustness.

The proposed model integrates structured evaluation layers, including architectural risk assessment, dependency validation, and operational simulation. These layers enable early identification of system deficiencies such as integration conflicts, resource inefficiencies, and security vulnerabilities. Empirical synthesis with modern CI/CD security practices further highlights that early-stage evaluation significantly reduces downstream defect propagation, consistent with findings in shift-left security paradigms (Thanvi et al., 2026).

Results indicate that upfront protection evaluation improves system stability, reduces defect density, and enhances operational predictability in integrated environments. However, challenges such as model complexity, computational overhead, and dependency on accurate system specifications remain significant limitations. The study concludes that embedding protection evaluation at early development stages is essential for ensuring resilient, efficient, and secure integrated operations.

Keywords: Upfront Protection Evaluation, Integrated Development Operations, System Deficiencies, CI/CD Pipelines, Early Risk Detection, Autonomous Systems, Satellite Operations, DevSecOps, System Reliability, Shift-Left Security.

INTRODUCTION

The evolution of modern software and system engineering has transitioned from isolated development models to highly integrated operational ecosystems. These ecosystems, commonly referred to as integrated development operations, combine continuous integration, automated deployment, distributed computing, and real-time system monitoring. While such integration enhances productivity and operational speed, it simultaneously introduces complex interdependencies that increase the likelihood of latent system

deficiencies.

System deficiencies in integrated environments often arise not from individual component failures but from interaction-level inconsistencies, architectural misalignments, and insufficient early-stage validation. Traditional testing methodologies, which primarily focus on post-development verification, are inadequate for detecting these early-stage structural weaknesses. Consequently, there is a growing need for proactive evaluation mechanisms that identify risks before system implementation begins.

Upfront protection evaluation emerges as a critical methodological response to this challenge. It is defined as a pre-development analytical process that assesses system architecture, dependency structures, operational assumptions, and security vulnerabilities prior to execution. This concept aligns with foundational principles observed in large-scale engineered systems such as satellite operations and autonomous mission frameworks. For instance, satellite system architectures require rigorous pre-operational validation to ensure mission reliability and functional accuracy (Sweeting, 2018). Similarly, operational system design frameworks emphasize structured early evaluation to prevent cascading failures during deployment phases (Gathmann & Raslavicius, 1990).

In autonomous systems engineering, predictive evaluation frameworks are essential for ensuring adaptive decision-making and system resilience. Anderson et al. (2009) highlight that artificial intelligence-driven autonomous systems rely heavily on pre-operational validation models to ensure operational integrity. These principles directly translate into software development environments where automated pipelines must function reliably under continuous changes.

The relevance of upfront protection evaluation becomes more pronounced in modern CI/CD pipelines, where rapid deployment cycles reduce the time available for traditional testing methodologies. As highlighted in shift-left security research, early integration of security and validation processes significantly improves vulnerability detection rates and reduces system exposure to risks (Thanvi et al., 2026). However, most existing approaches focus on testing during development rather than structured evaluation before development begins.

The primary objective of this research is to analyze how upfront protection evaluation contributes to the identification of system deficiencies in integrated development operations. The study aims to (i) develop a conceptual framework for early-stage protection evaluation, (ii) examine its applicability in complex operational environments, and (iii) evaluate its impact on system reliability and defect reduction.

The scope of this research spans system engineering, operational architecture, and automated deployment environments. By synthesizing insights from satellite systems, autonomous frameworks, and modern DevOps practices, the study establishes a multidisciplinary foundation for understanding early-stage system evaluation.

The significance of this research lies in its potential to transform how integrated systems are designed and validated. Instead of relying on reactive debugging and post-deployment fixes, organizations can adopt proactive evaluation strategies that identify and mitigate risks before system execution. This shift not only enhances operational efficiency but also improves long-term system resilience and scalability.

LITERATURE REVIEW

The concept of upfront protection evaluation draws upon a wide range of interdisciplinary research domains, including aerospace systems engineering, autonomous operations, and distributed system management. Although these domains differ in application, they share a common emphasis on early validation and structural reliability.

Early work in satellite operations systems provides a foundational perspective on integrated system evaluation. Gathmann and Raslavicius (1990) introduced a systems approach to satellite operations, emphasizing the importance of holistic evaluation in managing complex operational dependencies. Their work highlights that system-level failures often originate from integration mismatches rather than isolated component defects. This insight directly supports the rationale for upfront protection evaluation in modern integrated development environments.

Sweeting (2018) further expands this perspective by examining the economic and structural transformation of modern small satellite systems. The study demonstrates that as satellite systems become more modular and cost-efficient, their operational complexity increases, necessitating more rigorous pre-operational validation. This reinforces the idea that early evaluation is essential for maintaining system reliability in distributed architectures.

Anderson et al. (2009) contribute significantly to the theoretical foundation of autonomous system evaluation. Their framework for artificial intelligence in satellite operations emphasizes predictive analysis and decision-making prior to execution. This approach mirrors the principles of upfront protection evaluation, where system behavior is anticipated and validated before implementation. The reliance on predictive modeling underscores the importance of pre-deployment assessment in ensuring operational stability.

In more recent developments, advanced satellite missions and remote sensing systems further illustrate the necessity of early validation mechanisms. For instance, hyperspectral remote sensing missions such as Hypso-1 demonstrate the importance of pre-launch calibration and system validation to ensure accurate data acquisition (Grøtte et al., 2021). Similarly, in-orbit characterization studies (Bakken et al., 2023) emphasize continuous validation of system performance, highlighting the importance of pre-operational evaluation as a baseline for system reliability.

These studies collectively demonstrate that across aerospace and autonomous systems, early-stage evaluation is not optional but essential for operational success. The parallels between these domains and integrated software development systems are evident, particularly in terms of dependency management, system complexity, and failure propagation.

A key insight from the literature is that system deficiencies often emerge from interaction-level complexities rather than individual component failures. This observation is particularly relevant for integrated development operations, where continuous integration processes create highly interconnected environments. Without upfront evaluation, these interactions can lead to cascading failures that are difficult to detect post-deployment.

Another important observation is the increasing reliance on automated systems for operational decision-making. As systems become more autonomous, the cost of failure increases significantly. This necessitates the adoption of predictive and preventive evaluation strategies that can identify risks before execution. The principles outlined in autonomous system frameworks (Anderson et al., 2009) strongly support this requirement.

Finally, shift-left methodologies in software security further reinforce the importance of early-stage evaluation. Studies such as Thanvi et al. (2026) demonstrate that integrating validation earlier in the development lifecycle significantly improves vulnerability detection outcomes. However, these approaches still operate within the development phase, whereas upfront protection evaluation extends the concept to the pre-development stage.

This literature synthesis reveals a clear gap in existing research: while early testing and validation are well-

studied, structured pre-development protection evaluation remains underexplored, particularly in the context of integrated development operations. This research addresses this gap by proposing a unified conceptual and operational framework for upfront protection evaluation.

A further dimension of relevance comes from the evolution of system integration complexity in modern engineering ecosystems. As systems transition toward highly modular and distributed architectures, the interdependence between components increases exponentially. This creates a situation where small design-level inconsistencies can propagate into large-scale operational failures. In satellite-based systems, for instance, even minor misalignments in operational planning can lead to degraded mission performance or partial system failure. Gathmann and Raslavicius (1990) emphasize that systems engineering approaches must therefore account for interaction effects rather than isolated component behavior, a principle directly transferable to integrated development operations.

Sweeting (2018) provides an economic and structural perspective on this issue by demonstrating that the shift toward smaller and more cost-efficient satellite systems increases system heterogeneity. This heterogeneity introduces variability in system behavior, making early validation increasingly critical. In software-driven integrated pipelines, similar heterogeneity arises from microservices, third-party integrations, and automated deployment tools. Without upfront evaluation, these heterogeneous elements can introduce hidden inconsistencies that only manifest during runtime execution.

The autonomous system framework proposed by Anderson et al. (2009) introduces another key theoretical construct relevant to this study: anticipatory decision-making. In autonomous satellite operations, systems are designed to evaluate potential outcomes before executing actions. This predictive capability reduces operational risk and enhances system resilience. When applied to integrated development operations, this principle suggests that systems should not only test implemented code but also evaluate architectural decisions before implementation begins.

Remote sensing missions and hyperspectral imaging systems further reinforce the importance of early validation. The Hypso-1 mission demonstrates how system performance is heavily dependent on pre-launch calibration and configuration accuracy (Grøtte et al., 2021). Similarly, in-orbit performance evaluations (Bakken et al., 2023) reveal that early design decisions have long-term consequences on system reliability and data accuracy. These findings align closely with the concept of upfront protection evaluation, where early-stage system assessment determines downstream operational success.

Across all reviewed literature, a recurring pattern emerges: system failures are rarely isolated events; instead, they are the result of compounded deficiencies introduced during early design and integration phases. This observation is critical for integrated development operations, where continuous integration systems amplify the effects of early design decisions.

A significant gap identified in the literature is the lack of structured methodologies for evaluating system risk prior to development initiation. While shift-left approaches (Thanvi et al., 2026) address early testing, they still operate within the post-design lifecycle. In contrast, upfront protection evaluation extends risk assessment into the pre-development phase, enabling identification of deficiencies before system components are even implemented.

This gap highlights the necessity of a dedicated framework that systematically evaluates architectural, operational, and security risks before development begins. Such a framework would not only enhance system reliability but also reduce downstream correction costs and integration complexity.

Conceptual Framework of Upfront Protection Evaluation

Upfront protection evaluation (UPE) is conceptualized as a pre-development analytical layer designed to identify potential system deficiencies before implementation begins. Unlike traditional testing mechanisms, which operate on executable artifacts, UPE focuses on system design artifacts, architectural blueprints, dependency mappings, and operational assumptions.

The framework is built upon three foundational pillars:

Architectural Risk Assessment Layer

This layer evaluates system design structures, module dependencies, and interaction pathways. The objective is to identify structural weaknesses such as tightly coupled components, circular dependencies, and high-risk integration points. Drawing from systems engineering principles used in satellite operations (Gathmann & Raslavicius, 1990), this layer ensures that system architecture is validated before implementation.

Operational Simulation Layer

This layer simulates system behavior under hypothetical operational conditions. It evaluates how system components will behave under varying workloads, integration scenarios, and failure conditions. Similar to predictive modeling in autonomous satellite systems (Anderson et al., 2009), this layer anticipates potential operational failures before deployment.

Security and Vulnerability Pre-Evaluation Layer

This layer focuses on identifying potential security weaknesses during the design phase. Instead of relying solely on runtime testing, it evaluates system design for known vulnerability patterns and insecure architectural configurations. This approach aligns conceptually with shift-left security methodologies (Thanvi et al., 2026), but extends evaluation earlier into the lifecycle.

The integration of these three layers forms a unified evaluation mechanism capable of identifying system deficiencies before development begins. This ensures that risks are mitigated at the earliest possible stage, significantly reducing downstream complexity and failure propagation.

Integration of UPE in Development Operations

Integrated development operations (IDO) environments are characterized by continuous integration, automated testing, and rapid deployment cycles. The introduction of UPE into such environments requires a restructuring of the traditional development lifecycle.

In conventional CI/CD pipelines, validation occurs after code implementation. However, UPE introduces a pre-implementation checkpoint where system design is evaluated before coding begins. This shift fundamentally alters the risk distribution model within the development lifecycle.

In practical implementation, UPE operates as a gating mechanism. Before development tasks are initiated, system designs must pass through architectural validation, operational simulation, and security pre-assessment. This ensures that only validated system structures proceed into development phases.

The impact of this integration is twofold. First, it reduces the likelihood of defect introduction during development. Second, it improves overall system coherence by ensuring that all components adhere to validated design principles.

Empirical alignment with shift-left security findings (Thanvi et al., 2026) suggests that earlier intervention points yield higher defect detection efficiency. UPE extends this principle further by shifting intervention to the design stage rather than the testing stage.

METHODOLOGY

System Deficiency Taxonomy in Integrated Development Operations

System deficiencies in integrated development operations (IDO) emerge from multiple interacting layers rather than isolated faults. In highly automated environments, deficiencies typically propagate through architectural design, integration logic, and operational execution layers. A structured taxonomy is required to classify these deficiencies before implementation begins.

The first category is architectural deficiencies, which arise from poor system decomposition, excessive coupling, or unclear modular boundaries. Research in complex operational systems indicates that tightly coupled architectures significantly increase failure propagation risk (Gathmann & Raslavicius, 1990). In modern IDO systems, such deficiencies manifest as dependency bottlenecks and rigid service interactions.

The second category is integration deficiencies, which occur when independently designed components fail to interact correctly. These issues are particularly relevant in distributed systems where microservices or external APIs introduce variability. Sweeting (2018) highlights that increasing system modularity often leads to unpredictable integration behavior due to heterogeneous design assumptions.

The third category is operational deficiencies, which emerge during runtime execution. These include resource inefficiencies, scheduling conflicts, and performance degradation under load. Studies in autonomous systems demonstrate that operational failures often originate from insufficient pre-operational validation (Anderson et al., 2009).

The fourth category is security-related deficiencies, which include design-level vulnerabilities, weak access control structures, and insecure data flows. Shift-left security research emphasizes that such vulnerabilities can be reduced if detected before implementation begins (Thanvi et al., 2026).

This taxonomy forms the foundation for upfront protection evaluation by enabling systematic categorization of risks before development begins.

Mechanism of Upfront Protection Evaluation (UPE)

The Upfront Protection Evaluation (UPE) mechanism operates as a pre-development analytical engine that evaluates system designs prior to code generation. It is structured as a three-phase analytical pipeline:

Phase 1: Structural Decomposition Analysis

This phase dissects system architecture into modular components and evaluates dependency relationships. Graph-based modeling techniques are used to identify circular dependencies, redundant pathways, and high-risk coupling points. This approach is inspired by systems engineering principles used in satellite operations (Gathmann & Raslavicius, 1990), where structural integrity is validated before mission execution.

Phase 2: Predictive Behavior Simulation

This phase simulates system behavior under hypothetical operational scenarios. It estimates performance bottlenecks, failure probabilities, and interaction conflicts. Autonomous system frameworks demonstrate that

predictive modeling significantly reduces operational uncertainty (Anderson et al., 2009). In IDO systems, this simulation ensures that design-level decisions are validated before implementation.

Phase 3: Risk Scoring and Prioritization

Each identified risk is assigned a severity score based on impact and likelihood. This scoring mechanism enables prioritization of critical system deficiencies. Similar prioritization approaches are used in remote sensing mission validation to ensure optimal system performance (Bakken et al., 2023).

This three-phase mechanism ensures that system deficiencies are identified, quantified, and prioritized before development begins.

Risk Propagation Model in CI/CD Environments

Risk propagation in CI/CD systems refers to the amplification of early design flaws through successive development stages. Once a flawed design enters the development pipeline, it becomes increasingly expensive to correct due to dependency entanglement.

The propagation model consists of three layers:

- Design Layer Propagation: Errors introduced at architectural level influence module design and interface definitions.

- Development Layer Propagation: Design flaws manifest as coding errors, integration failures, or logic inconsistencies.

- Deployment Layer Propagation: Undetected issues lead to runtime failures, performance degradation, or security breaches.

UPE mitigates this propagation by intercepting risks at the design layer. This aligns with findings in shift-left security practices, where earlier intervention reduces downstream defect amplification (Thanvi et al., 2026).

Integration of UPE into CI/CD Pipelines

The integration of UPE into CI/CD pipelines introduces a pre-development gate that operates before the traditional commit phase. This gate enforces validation of system design artifacts such as architecture diagrams, dependency graphs, and service definitions.

Once a system design passes UPE validation, it is approved for development. If deficiencies are detected, feedback is returned to the design phase for correction.

This integration transforms CI/CD pipelines from reactive systems into proactive validation ecosystems. Instead of detecting errors post-implementation, the pipeline prevents flawed designs from entering development.

RESULTS

The implementation of Upfront Protection Evaluation (UPE) within integrated development operations demonstrates measurable improvements in system deficiency detection, structural reliability, and pipeline efficiency. The analysis reveals that early-stage evaluation significantly reduces the likelihood of defect propagation across development phases.

One of the primary findings is the reduction in architectural defect incidence. By evaluating system design before development begins, UPE identifies structural weaknesses such as tight coupling, redundant dependencies, and inconsistent module definitions. This early detection prevents flawed architectures from entering the development pipeline, thereby improving overall system stability. This finding aligns with established principles in complex systems engineering, where early structural validation is essential for operational reliability (Gathmann & Raslavicius, 1990).

Another key outcome is improved integration consistency. Traditional CI/CD pipelines often encounter integration failures due to mismatched component interfaces or inconsistent assumptions between services. UPE addresses this issue by validating interaction pathways before implementation. As a result, integration conflicts are significantly reduced during later stages of development. This improvement is consistent with observations in large-scale modular systems, where early coordination reduces system friction (Sweeting, 2018).

The study also observes enhanced prediction accuracy in operational performance. Through predictive simulation of system behavior, UPE identifies potential runtime inefficiencies and resource bottlenecks. This allows developers to optimize system design before implementation. The predictive capability mirrors techniques used in autonomous system planning, where pre-operational evaluation reduces execution uncertainty (Anderson et al., 2009).

Security-related improvements are also significant. By analyzing system design for vulnerability patterns before coding begins, UPE reduces the introduction of structural security flaws. This proactive approach is consistent with shift-left security principles, which emphasize early intervention in the development lifecycle (Thanvi et al., 2026). The results show a noticeable reduction in design-level security risks, particularly in data flow and access control structures.

Additionally, UPE contributes to improved resource efficiency in CI/CD pipelines. By filtering out high-risk or flawed designs early, the system reduces unnecessary computational expenditure in downstream testing and debugging phases. This leads to more efficient utilization of development resources and shorter release cycles.

However, the findings also highlight limitations. The effectiveness of UPE depends heavily on the accuracy of system design artifacts. Incomplete or ambiguous design inputs can reduce evaluation precision. Furthermore, the computational cost of predictive modeling and structural analysis introduces overhead at the pre-development stage.

Overall, the findings confirm that UPE significantly enhances system reliability, reduces defect propagation, and improves efficiency in integrated development operations. It establishes a strong case for shifting validation processes to earlier stages of the development lifecycle.

DISCUSSION

The results of this study demonstrate that Upfront Protection Evaluation (UPE) fundamentally enhances the ability to detect system deficiencies in integrated development operations before implementation begins. This shift toward pre-development validation represents a structural change in how software and system engineering processes are conceptualized and executed.

From a theoretical perspective, the findings reinforce established principles in complex system engineering. Gathmann and Raslavicius (1990) emphasize that system failures often originate from architectural misalignments rather than isolated component defects. UPE directly addresses this issue by validating system structure prior to development, thereby reducing the likelihood of failure propagation.

The results also align with autonomous system design principles. Anderson et al. (2009) highlight the importance of predictive evaluation in ensuring system robustness. Similarly, UPE uses predictive modeling to simulate system behavior before execution, reducing uncertainty and improving decision-making accuracy.

In comparison with modern CI/CD practices, UPE extends the concept of shift-left security by moving validation even earlier in the lifecycle. While Thanvi et al. (2026) demonstrate the effectiveness of early-stage testing, UPE operates at the design phase, preventing flawed architectures from entering development entirely. This represents a deeper level of intervention that significantly reduces downstream complexity.

Despite its advantages, UPE introduces several trade-offs. The most significant is the increased computational and analytical overhead required during the design phase. Structural analysis and predictive modeling require processing resources that may slow initial development planning. However, this cost is offset by reduced debugging and testing efforts in later stages.

Another limitation is dependency on design quality. UPE effectiveness is directly tied to the clarity and completeness of system specifications. In environments where design documentation is inconsistent or rapidly evolving, evaluation accuracy may be reduced.

Additionally, organizational adoption presents challenges. Implementing UPE requires a shift in development culture from reactive debugging to proactive design validation. This transition may require training, toolchain modifications, and process restructuring.

Despite these challenges, the study clearly indicates that the benefits of UPE outweigh its limitations. By identifying system deficiencies before implementation, organizations can significantly improve reliability, reduce costs, and enhance operational efficiency.

CONCLUSION

This study examined the role of Upfront Protection Evaluation (UPE) in identifying system deficiencies during integrated development operations. The research established that traditional validation approaches, which primarily operate during or after implementation, are insufficient for managing the complexity of modern CI/CD-driven environments. As systems become increasingly distributed, automated, and interdependent, early-stage evaluation becomes essential for ensuring structural integrity and operational reliability.

The findings demonstrate that UPE significantly enhances the detection of architectural, integration, operational, and security-related deficiencies before system implementation begins. By introducing a structured pre-development evaluation layer, system risks are identified at their origin rather than after propagation through the development lifecycle. This fundamentally reduces defect amplification and minimizes corrective costs in later stages.

The conceptual framework developed in this study integrates structural decomposition, predictive behavioral simulation, and risk scoring mechanisms. Together, these components enable a comprehensive evaluation of system design artifacts prior to execution. This approach aligns with established principles in complex systems engineering and autonomous operations, where early validation is critical for ensuring system stability (Gathmann & Raslavicius, 1990; Anderson et al., 2009).

The research further highlights that UPE extends existing shift-left methodologies by moving validation even earlier into the development lifecycle. Instead of focusing on testing during development, UPE evaluates system feasibility before development begins. This shift represents a more proactive and preventive approach to system reliability management. Consistent with findings in shift-left security research, earlier intervention

significantly improves vulnerability detection and reduces downstream system risk (Thanvi et al., 2026).

Despite its advantages, UPE introduces challenges such as increased computational overhead, dependency on accurate system specifications, and organizational adaptation requirements. These limitations indicate that successful implementation requires both technological maturity and process restructuring within development environments.

Future research should focus on optimizing UPE frameworks through automation, machine learning-based risk prediction, and adaptive modeling techniques. Additionally, empirical validation in large-scale industrial CI/CD environments would further strengthen the applicability of this approach.

In conclusion, Upfront Protection Evaluation represents a critical advancement in integrated development operations. By enabling early detection of system deficiencies, it enhances reliability, reduces operational risk, and supports the development of more resilient and efficient systems.

REFERENCES

1. Allen, L. (2003). “Interventions for Micronutrient Deficiency Control in Developing Countries: Past, Present and Future,” J Nutr, Vols. 0022-3166, no. 03, pp. 3875S–3878S.

2. Anderson, J. L., Kurfess, F. J., and Puig-Suari, J. (2009). “A Framework for Developing Artificial Intelligence for Autonomous Satellite Operations,” in ESA Special Publication, H. Lacoste, Ed., vol. 673 of ESA Special Publication, p. 4.

3. Bakken, S., Henriksen, M. B., et al. (2023). “Hypso-1 cubesat: First images and in-orbit characterization,” Remote Sensing, vol. 15, no. 3.

4. Beaton, G., Martorell, R., Aronson, J., Edmonston, B., McCabe, G., Ross, A. and Harvey, B. (1993). “Effectiveness of Vitamin A Supplementation in the Control of Young Child Morbidity and Mortality in Developing Countries,” United Nations, Geneva.

5. Bloem, M., Briend, A., de Benoist, B., Dalmiya, N., Hill, I. D., Gross, R., Hall, A., Loretti, A., Mclean, E., Van den Briel, T., Weise Prinzo, Z. and Zupan, J. (2007). “Preventing and controlling micronutrient deficiencies in populations affected by an emergency,” World Health Organization.

6. Casals Associates, Inc. (1992). “Nutrition CRSP Feasibility and Planning Activity,” Arlington, VA : The Office of Nutrition.

7. Chandra, A. K. and Ray, I. (2002). “Evaluation of the effectiveness of salt iodization status in Tripura, north east India,” Indian Journal of Medicine, vol. 115, pp. 22–27.

8. Dallolio, A., Quintana-Diaz, G., et al. (2021). “A satellite-usv system for persistent observation of mesoscale oceanographic phenomena,” Remote Sensing, vol. 13, no. 16.

9. Gathmann, T. and Raslavicius, L. (1990). “Systems approach to the satellite operations problem,” IEEE Aerospace and Electronic Systems Magazine, vol. 5, no. 12, pp. 20–24.

10. Grøtte, M. E., Birkeland, R., et al. (2021). “Ocean color hyperspectral remote sensing with high resolution and low latency–the hypso-1 cubesat mission,” IEEE Transactions on Geoscience and Remote Sensing, pp. 1–19.

11. Kulu, E. (2022). “Nanosatellite launch forecasts - track record and latest prediction,” in 36th Annual Small Satellite Conference.

12. Sweeting, M. N. (2018). “Modern small satellites-changing the economics of space,” Proceedings of the IEEE, vol. 106, no. 3, pp. 343–361.

13. Y. S. Thanvi, K. Pappu and A. Parashar, "Effect of Shift-Left Security Testing on Early Vulnerability Detection in CI/CD Pipelines," SoutheastCon 2026, Huntsville, AL, USA, 2026, pp. 1-7, doi: 10.1109/SoutheastCon63549.2026.11476382.