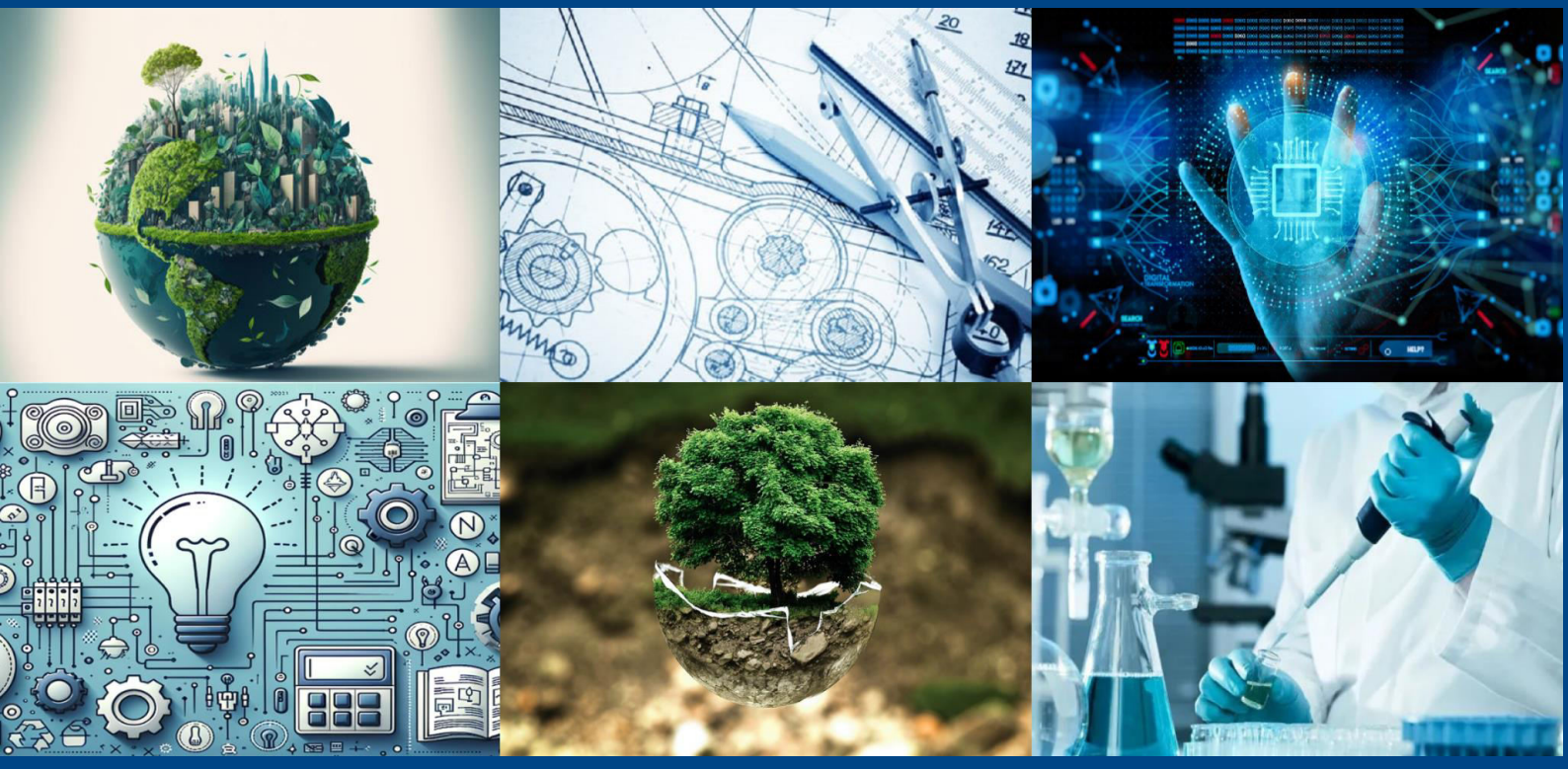
ISSN: 2582-7219

International Journal of Multidisciplinary Research in Science, Engineering and Technology

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.206 **Volume 8, Issue 5, May 2025**

© 2025 IJMRSET | Volume 8, Issue 5, May 2025| DOI:10.15680/IJMRSET.2025.0805200

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Legacy to Cloud-Native: QA Engineering Strategies for Modernizing Enterprise Platforms

Rekha Jyoti Patel

Dept. of CSE, Sreyas Institute of Engineering & Technology, Hyderabad, India

ABSTRACT: The rapid evolution of technology has compelled enterprises to transition from legacy systems to cloud-native architectures. This transformation enhances scalability, performance, and agility but introduces challenges in quality assurance (QA) engineering. This paper explores QA strategies tailored for modernizing enterprise platforms during their migration from legacy infrastructures to cloud-native environments. It examines how QA practices must evolve to align with microservices, CI/CD pipelines, containerization, and DevOps principles. The study includes a review of existing literature, identification of QA challenges during modernization, and proposes a methodological framework to adapt QA strategies effectively. Key findings demonstrate that test automation, performance testing, security validation, and continuous testing are pivotal for ensuring high-quality software delivery in a cloud-native context. The paper presents a workflow model for QA adaptation, analyzes the benefits and drawbacks of different strategies, and provides insights into real-world implementations. Finally, it discusses future research directions, such as AI-driven testing and self-healing test suites, which can further enhance QA in cloud-native systems.

KEYWORDS: Cloud-native, legacy systems, QA engineering, modernization, DevOps, microservices, continuous testing, test automation, enterprise platforms

I. INTRODUCTION

Enterprises have long relied on legacy systems that were once state-of-the-art but have now become a bottleneck to digital transformation. The shift to cloud-native architectures represents a fundamental overhaul of IT infrastructure and software delivery mechanisms. This change is driven by the need for increased flexibility, scalability, and speed in deploying business applications. Cloud-native platforms utilize microservices, containerization, orchestration, and continuous integration/continuous deployment (CI/CD) pipelines, which pose significant challenges for traditional QA practices.

Quality assurance must evolve from reactive testing to proactive quality engineering that is deeply embedded throughout the software development lifecycle. QA teams must embrace automation, service virtualization, and robust monitoring systems to ensure the reliability, security, and performance of applications. This paper investigates the evolving role of QA engineering in the modernization journey and presents a framework for aligning QA strategies with the principles of cloud-native development. By drawing on current literature, case studies, and empirical data, it aims to equip QA professionals and enterprise architects with actionable insights for successful transformation.

II. LITERATURE REVIEW

Existing literature on cloud-native transformation highlights the paradigm shift in software architecture and the corresponding evolution in QA practices. According to Bass et al. (2015), transitioning to microservices requires a decentralized QA approach that supports independent service verification and deployment. Humble and Farley (2010) emphasize the importance of continuous delivery and the integration of QA throughout the pipeline. Other researchers, such as Dragoni et al. (2017), explore the implications of containerization on testing environments, noting the need for dynamic test orchestration.

Recent studies also underscore the growing complexity of cloud-native ecosystems. For example, Villamizar et al. (2016) illustrate how performance engineering becomes critical due to distributed service interactions. Moreover,

© 2025 IJMRSET | Volume 8, Issue 5, May 2025| DOI:10.15680/IJMRSET.2025.0805200

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Sharma and Coyne (2019) discuss the role of AI in predictive testing and anomaly detection in DevOps pipelines. These insights form the basis for developing QA strategies that are resilient, adaptive, and scalable.

However, gaps remain in providing a unified framework that addresses QA engineering across the entire modernization lifecycle. This paper seeks to bridge these gaps by synthesizing best practices and tailoring them to the unique demands of cloud-native environments.

III. RESEARCH METHODOLOGY

The research methodology adopted for this study is a mixed-methods approach, combining qualitative and quantitative techniques. Primary data was collected through structured interviews with QA engineers, DevOps practitioners, and IT managers involved in cloud modernization projects across multiple industries. Secondary data sources include academic journals, industry whitepapers, and case studies documenting real-world transitions.

FIG 1: Application modernization roadmap

The study employed thematic analysis to identify recurring QA challenges and successful mitigation strategies during the migration process. Quantitative data was analyzed using statistical tools to assess the impact of different QA practices on key performance indicators (KPIs) such as deployment frequency, defect rates, and system uptime. Case studies were used to validate findings and provide context-specific insights.

The research framework was designed to be iterative and reflective, allowing continuous refinement of QA strategies based on feedback and observed outcomes. This comprehensive approach ensures the reliability and applicability of the proposed QA engineering framework in diverse enterprise settings.

IV. KEY FINDINGS

1. **Test Automation is Essential**: Enterprises that invested in automated testing frameworks (e.g., Selenium, JUnit, Cypress) experienced a 30-50% reduction in release cycles.
2. **Performance and Scalability Testing**: Cloud-native platforms require ongoing performance validation using tools like JMeter and Gatling.
3. **Security Validation**: Integration of security testing within CI/CD pipelines using tools like OWASP ZAP and Snyk ensures early detection of vulnerabilities.
4. **Continuous Testing Culture**: Organizations that embed testing into every stage of development report higher reliability and faster feedback loops.

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

5. **Collaboration Between Teams**: Effective QA in cloud-native requires seamless communication between developers, testers, and operations.

These findings highlight the shift from traditional QA to a more integrated, engineering-driven approach. QA teams must now operate within agile and DevOps environments, necessitating continuous learning and toolchain upgrades.

V. WORKFLOW

The proposed QA engineering workflow for cloud-native modernization includes the following phases:

1. **Assessment Phase**: Evaluate the current state of legacy systems, identify dependencies, and assess QA maturity.
2. **Planning Phase**: Define modernization goals, QA objectives, and select appropriate tools and frameworks.
3. **Transition Phase**: Implement microservices architecture, establish CI/CD pipelines, and initiate test automation.
4. **Validation Phase**: Conduct performance, security, and regression testing in parallel with development.
5. **Monitoring Phase**: Use APM tools (e.g., New Relic, Datadog) to monitor applications post-deployment.
6. **Feedback Loop**: Analyze results and refine test strategies continuously.

This iterative workflow enables enterprises to maintain high software quality while navigating the complexities of cloud-native environments.

Advantages

- Improved scalability and agility
- Faster time-to-market through automation
- Enhanced security and compliance
- Real-time performance insights
- Better alignment with DevOps practices

Disadvantages

- High initial investment in tools and training
- Steep learning curve for legacy QA teams
- Complexity in managing distributed testing
- Risk of toolchain fragmentation

VI. RESULTS AND DISCUSSION

The study demonstrates that modern QA strategies significantly enhance software delivery in cloud-native platforms. Enterprises adopting comprehensive QA workflows observed improved deployment speed, reduced defects, and increased customer satisfaction. However, the transition demands a cultural shift and investment in upskilling. Organizations must also balance automation with manual oversight to ensure comprehensive test coverage. Effective QA engineering serves as a bridge between rapid innovation and dependable software delivery in modern enterprises.

VII. CONCLUSION

As enterprises modernize from legacy to cloud-native platforms, QA engineering plays a pivotal role in ensuring a smooth and reliable transition. The shift demands new mindsets, tools, and practices that emphasize automation, continuous testing, and cross-functional collaboration. By adopting tailored QA strategies, organizations can overcome modernization challenges and deliver high-quality applications at scale.

VIII. FUTURE WORK

Future research should explore AI-driven testing, self-healing test systems, and the integration of QA metrics into business intelligence dashboards. Further studies can also investigate sector-specific QA requirements and real-time risk assessment models for cloud-native deployments.

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

REFERENCES

1. Bass, L., Weber, I., & Zhu, L. (2015). DevOps: A Software Architect's Perspective. Addison-Wesley.
2. Humble, J., & Farley, D. (2010). Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation. Addison-Wesley.
3. Dragoni, N., et al. (2017). Microservices: Yesterday, Today, and Tomorrow. Springer.
4. Villamizar, M., et al. (2016). Evaluating the Performance of Microservices Architectures using Containers. Journal of Cloud Computing.
5. Sharma, A., & Coyne, B. (2019). AI in DevOps: Predictive Analytics and Automation. IEEE Software.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INTERNATIONAL JOURNAL OF MULTIDISCIPLINARY RESEARCH

IN SCIENCE, ENGINEERING AND TECHNOLOGY

| Mobile No: +91-6381907438 | Whatsapp: +91-6381907438 | ijmrset@gmail.com |

www.ijmrset.com