INTERNATIONAL JOURNAL OF MULTIDISCIPLINARY EVOLUTIONARY RESEARCH

A Risk-Aware AI Framework for Automated Testing and Quality Assurance in Core Banking Systems

Sai Kumar Gunda
Software Quality Analyst, Tata Consultancy Services Ltd. (Client: Citi Bank), Long Island City, New York, United States

* Corresponding Author: **Sai Kumar Gunda**

Article Info

P-ISSN: 3051-3502
E-ISSN: 3051-3510
Volume: 05
Issue: 01
January - June 2024
Received: 15-12-2023
Accepted: 17-01-2024
Published: 19-02-2024
Page No: 117-120

Abstract
Core banking platforms operate under stringent availability, integrity, and regulatory expectations while evolving rapidly through continuous delivery, vendor updates, and complex integration landscapes. Traditional quality assurance approaches struggle to maintain defect containment, security assurance, and audit readiness as test suites scale and system dependencies grow. This paper proposes a risk-aware AI framework that continuously converts operational, cybersecurity, compliance, and model risks into test strategy decisions, enabling automated test selection, prioritization, and governance aligned with core banking constraints. The framework integrates (i) risk-driven change impact analysis, (ii) AI-based defect propensity and failure propagation signals, (iii) policy-aware test orchestration with verifiable control mapping, and (iv) model governance primitives for explainability, robustness, and bias monitoring. A simulation-based evaluation illustrates how risk-weighted prioritization improves time-to-detection and risk-weighted recall under fixed execution budgets. The approach is designed for hybrid stacks typical of banks (batch and online, legacy and microservices) and emphasizes audit-friendly evidence generation, reducing unknown-risk exposure while improving release velocity.

DOI: https://doi.org/10.54660/IJMER.2024.5.1.117-120

Keywords: Core banking, risk-aware testing, AI-driven QA, automated testing, defect prediction, compliance engineering, operational resilience, secure SDLC

1. Introduction

Core banking systems are engineered for continuity under stress, and operational resilience principles increasingly shape how banks plan, test, and recover from disruptions.[9] Regulatory expectations for ICT risk management and incident readiness further raise the bar for software quality evidence, especially when delivery pipelines change frequently.[17] In parallel, regression suites often grow faster than execution capacity, creating a structural gap between tests we have and tests we can afford to run during time-critical releases.

Two trends create a practical opportunity: automation economics (measurable time and cost savings) and AI signals that can focus QA effort where it matters most.[5] However, applying AI in high-stakes systems requires more than accuracy; it requires risk-aware decisioning, governance, and traceable evidence that can be defended in reviews and audits.

2. Background and Problem Statement

Software testing in regulated environments must be systematic, repeatable, and evidence producing, which aligns naturally with standardized testing concepts and definitions.[16] Yet core banking introduces constraints that complicate classical test optimization:

- High blast radius: defects in ledger posting, payments, or interest calculation propagate across products and channels.
- Mixed architecture: online services coexist with batch jobs, MQ-driven flows, mainframe or vendor modules, and external payment networks.

- Regulatory and audit pressure: releases require demonstrable controls and traceability, not only defect counts.
- Security coupling: functional changes can modify attack surfaces and data handling paths.

The resulting QA problem is not simply find more defects, but maximize risk reduction per unit test time while producing auditable proof.

3. Related Work

Defect prediction has shown value as a prioritization signal for QA, with comparative model studies demonstrating that different learners behave differently under software-specific feature distributions.[7] Complementary analyses further illustrate how model choice impacts fault prediction effectiveness across common classifiers used in applied software engineering contexts.[18] Architecture-centered decision intelligence approaches have argued for integrating defect prediction, automated testing, and governance into an end-to-end lifecycle framework rather than treating them as isolated optimizations.[13]

Because core banking is a high-stakes domain, it is also instructive to look at adjacent safety-critical sectors where AI adoption has been tied to clinical practice under strong accountability constraints.[20] Finally, security and cybersecurity convergence discussions highlight that QA must treat security assurance as a first-class quality dimension rather than an external gate.[3]

4. Framework Overview

We propose RAAIT-QA (Risk-Aware AI Testing for Quality Assurance), a closed-loop framework that continuously transforms risk context into test decisions and evidence artifacts.

4.1. Risk and AI Governance Anchors

RAAIT-QA aligns AI system risk thinking with a structured taxonomy and measurement approach that is compatible with established AI risk management guidance.[11] It also applies risk management guidance focused specifically on AI, including lifecycle risk identification, treatment, and monitoring.[2] Governance implications of organizational AI use are addressed through IT governance guidance that frames oversight, accountability, and decision rights.[14] Finally, the framework can be institutionalized using an AI management system approach to formalize roles, controls, and continuous improvement.[8]

4.2. Terminology and System Decomposition

To standardize how teams describe AI components (models, data, inference, monitoring) and avoid ambiguity in audits, RAAIT-QA adopts consistent AI vocabulary and concept framing.[4] The framework also maps AI-enabled functions to system lifecycle building blocks commonly used to structure machine-learning-based systems.[23]

5. Risk Model and Decision Layer

RAAIT-QA uses a risk-to-test decision layer that outputs an execution plan under constraints (time budget, environment availability, change freeze windows, and release criticality).

5.1. Risk Vector Construction

For each change-set c, we compute a risk vector:

r(c) = [r_op, r_sec, r_comp, r_data, r_model]

where r_op is operational risk (blast radius, dependency depth, recovery complexity), r_sec is cybersecurity risk (attack surface shift and sensitive data paths), r_comp is compliance risk (payments, privacy boundaries, reporting), r_data is data integrity and reconciliation risk, and r_model is AI model risk (uncertainty, drift, explainability gaps).

5.2. Test Value Scoring

Each test t receives a value score combining predicted failure likelihood, criticality coverage, and evidence yield:

V(t, c) = alpha * P(fail | t, c) + beta * Crit(t) + gamma * Evidence(t)

subject to the runtime budget constraint sum(runtime(t)) <= B. The optimizer selects and orders tests to maximize total value with policy constraints.

5.3. Governance Requirements for Explainability, Bias, and Robustness

RAAIT-QA requires that prioritization decisions be explainable to reviewers and auditors through transparent model documentation and rationale capture practices.[27] Because prioritization models can systematically under-test certain modules or customer flows, the framework monitors and mitigates bias risks using recognized algorithmic bias considerations and evaluation guidance.[28] Bias analysis is treated as an AI risk artifact that is periodically re-evaluated as data and systems evolve.[26] Robustness expectations for neural components are operationalized through robustness assessment practices, especially when models influence safety-relevant release decisions.[19]

6. Security, Compliance, and Evidence Engineering

Core banking QA must simultaneously reduce defects and strengthen security posture, particularly where software quality intersects with cyber risk and public trust.[15] RAAIT-QA embeds information security management into QA evidence pipelines via an ISMS-aligned approach to controls, traceability, and continual improvement.[24] Control selection and verification evidence are aligned with widely used security control guidance so that test evidence can be mapped to control objectives and review checklists.[25]

6.1. Secure SDLC and Supply Chain Considerations

RAAIT-QA integrates secure software development practices into CI pipelines so that quality signals include security-relevant artifacts by default.[1] It also treats third-party and open-source dependencies as risk contributors, incorporating supplier and component risk practices to reduce systemic exposure through upstream weaknesses.[21]

6.2. Application Security Verification and Common Weakness Coverage

To ensure functional changes do not introduce common exploit paths, the framework maps test evidence to widely

recognized categories of application risk.[6] Verification evidence is structured to align with application security verification requirements, enabling repeatable assurance and consistent release criteria.[22] For payment-related paths, RAAIT-QA includes compliance-aware testing evidence that supports payment security control expectations.[12]

6.3. Operational Resilience Alignment

Beyond cybersecurity, RAAIT-QA treats service continuity testing as a core QA objective, reflecting the broader importance of resilient digital infrastructure in modern economies.[10]

7. Implementation in Core Banking Environments

RAAIT-QA integrates with typical bank delivery stacks: CI runners, service virtualization, masked or synthetic test data, environment reservation systems, and evidence stores. Key implementation elements include:

Change intelligence: build dependency graphs from service catalogs, message flows, batch DAGs, and database lineage.

AI signal extraction: compute defect propensity and failure propagation signals from code churn, configuration deltas, historical failures, and runtime telemetry.

Policy-aware orchestration: enforce must-run suites for ledger and payments, apply environment constraints, and preserve segregation-of-duty requirements.

Evidence generation: auto-produce traceable artifacts (test runs, coverage-to-control mapping, approval records, and model monitoring snapshots).

8. Evaluation via Simulation

Because production core banking data is often restricted, we provide a simulation-based evaluation that reflects common constraints: large suites, limited execution budgets, and uneven risk distribution across modules.

8.1. Setup

We simulated a suite of 1200 tests across 180 modules and evaluated 400 change events under a fixed budget equal to 30 percent of full-suite runtime. We compared: (i) Baseline: random test ordering within the budget; and (ii) RAAIT-QA: risk-weighted prioritization using predicted failure likelihood and criticality coverage.

8.2. Results

Table 1: summarizes simulation outcomes under the fixed budget (30 percent of full-suite time).

Metric	Baseline	RAAIT-QA
Detect@budget (probability of catching >= 1 failure)	0.7375	0.8525
Median time-to-first-failure (seconds)	1662.07	201.57
Mean recall@budget (fraction of failing tests caught)	0.2767	0.6109
Mean risk-weighted recall@budget	0.2857	0.6454

The results indicate that risk-aware ordering improves both speed (earlier failure discovery) and depth (capturing a larger fraction of failures), particularly for higher-criticality areas.

9. Discussion and Threats to Validity

Auditability versus adaptivity: AI-driven prioritization must not become a black box. RAAIT-QA addresses this through transparent rationale capture, control mappings, and periodic governance review artifacts.

Data constraints: banks may lack unified defect labels across legacy systems; the framework supports weak supervision and incremental labeling strategies.

Model drift: changing product mixes, infrastructure, and vendor upgrades can shift failure modes; RAAIT-QA requires continuous monitoring and refresh triggers.

Simulation limits: the evaluation illustrates directional benefit under realistic constraints, but production outcomes will depend on data quality, environment fidelity, and organizational adherence to governance processes.

10. Conclusion

This paper presented RAAIT-QA, a risk-aware AI framework that converts operational, security, compliance, and model risks into automated testing and QA decisions for core banking systems. By integrating defect propensity signals, policy-aware orchestration, and governance-grade evidence engineering, the approach targets maximum risk reduction per unit test time while improving audit readiness. The simulation results suggest substantial gains in time-to-detection and risk-weighted recall under constrained budgets. Future work includes multi-bank benchmarking with privacy-preserving learning, richer failure propagation models, and standardized audit evidence schemas for AI-assisted QA.

References

1. National Institute of Standards and Technology. Secure software development framework (SSDF) version 1.1: recommendations for mitigating the risk of software vulnerabilities. Gaithersburg (MD): NIST; 2022. (NIST Special Publication 800-218).
2. International Organization for Standardization, International Electrotechnical Commission. ISO/IEC 23894:2023 Information technology — Artificial intelligence — Guidance on risk management. Geneva: ISO/IEC; 2023.
3. Pittala SK, Ashok VKC. A new era in security: bridging information security and cybersecurity. Int J Multidiscip Futur Dev. 2023;4(1):69-72. doi: 10.54660/IJMFD.2023.4.1.69-72.
4. International Organization for Standardization, International Electrotechnical Commission. ISO/IEC 22989:2022 Information technology — Artificial intelligence — Artificial intelligence concepts and terminology. Geneva: ISO/IEC; 2022.
5. Kacheru G, Bajjuru R, Arthan N. The ROI of software automation: measuring time and cost savings. Int J Commun Netw Inf Secur. 2023;15(4):774-85.
6. OWASP Foundation. OWASP Top 10: the ten most critical web application security risks. 2021.
7. Gunda SK. Comparative analysis of machine learning models for software defect prediction. In: 2024 International Conference on Power, Energy, Control and Transmission Systems (ICPECTS), Chennai, India.

2024. p. 1-6. doi: 10.1109/ICPECTS62210.2024.10780167.
8. International Organization for Standardization, International Electrotechnical Commission. ISO/IEC 42001:2023 Information technology — Artificial intelligence — Management system. Geneva: ISO/IEC; 2023.
9. Basel Committee on Banking Supervision. Principles for operational resilience. Basel: BCBS; 2021.
10. Ashok VKC. Cybersecurity for smart infrastructure and public utilities. Int J Multidiscip Res Growth Eval. 2023;4(2):947-9. doi: 10.54660/IJMRGE.2023.4.2.947-949.
11. National Institute of Standards and Technology. Artificial intelligence risk management framework (AI RMF 1.0). Gaithersburg (MD): NIST; 2023. (NIST AI 100-1).
12. Payment Card Industry Security Standards Council. PCI DSS v4.0. 2022.
13. Sivva SD, Thalakanti RR, Bandari SSG, Yettapu SDR. AI-driven decision intelligence for agile software lifecycle governance: an architecture-centered framework integrating machine learning defect prediction and automated testing. Int J Eng Technol Comput Sci Inf Technol. 2023 Dec 30;4(4):167-72. Available from: https://www.ijetcsit.org/index.php/ijetcsit/article/view/554.
14. International Organization for Standardization, International Electrotechnical Commission. ISO/IEC 38507:2022 Information technology — Governance of IT — Governance implications of the use of artificial intelligence by organizations. Geneva: ISO/IEC; 2022.
15. Pittala SK. Cybersecurity and online safety: a critical asset in the information era. J Front Multidiscip Res. 2023;4(1):576-9. doi: 10.54660/JFMR.2023.4.1.576-579.
16. International Organization for Standardization, International Electrotechnical Commission, Institute of Electrical and Electronics Engineers. ISO/IEC/IEEE 29119-1:2022 Software and systems engineering — Software testing — Part 1: General concepts. Geneva: ISO/IEC/IEEE; 2022.
17. European Union. Regulation (EU) 2022/2554 on digital operational resilience for the financial sector (DORA). 2022.
18. Gunda SK. Fault prediction unveiled: analyzing the effectiveness of random forest, logistic regression, and KNeighbors. In: 2024 2nd International Conference on Self Sustainable Artificial Intelligence Systems (ICSSAS), Erode, India. 2024. p. 107-13. doi: 10.1109/ICSSAS64001.2024.10760620.
19. International Organization for Standardization, International Electrotechnical Commission. ISO/IEC TR 24029-1:2021 Artificial intelligence — Assessment of the robustness of neural networks — Part 1: Overview. Geneva: ISO/IEC; 2021.
20. Kacheru G. Revolutionizing healthcare: the role of artificial intelligence in clinical practice. J Comput Anal Appl. 2023;31(4):1546-54. Available from: https://eudoxuspress.com/index.php/pub/article/view/3270.
21. National Institute of Standards and Technology. Cybersecurity supply chain risk management practices for systems and organizations. Gaithersburg (MD): NIST; 2022. (NIST Special Publication 800-161 Revision 1).
22. OWASP Foundation. Application security verification standard (ASVS) v4.0.3. 2021.
23. International Organization for Standardization, International Electrotechnical Commission. ISO/IEC 23053:2022 Framework for artificial intelligence (AI) systems using machine learning (ML). Geneva: ISO/IEC; 2022.
24. International Organization for Standardization, International Electrotechnical Commission. ISO/IEC 27001:2022 Information security, cybersecurity and privacy protection — Information security management systems — Requirements. Geneva: ISO/IEC; 2022.
25. International Organization for Standardization, International Electrotechnical Commission. ISO/IEC 27002:2022 Information security, cybersecurity and privacy protection — Information security controls. Geneva: ISO/IEC; 2022.
26. International Organization for Standardization, International Electrotechnical Commission. ISO/IEC TR 24027:2021 Information technology — Artificial intelligence — Bias in AI systems and AI aided decision making. Geneva: ISO/IEC; 2021.
27. Institute of Electrical and Electronics Engineers. IEEE Std 7001-2021 Standard for transparency of autonomous systems. 2021.
28. Institute of Electrical and Electronics Engineers. IEEE Std 7003-2022 Standard for algorithmic bias considerations. 2022.