International Journal of Scientific Research and Engineering Development-– Volume 9 Issue 6, Nov – Dec 2026
Available at www.ijsred.com

RESEARCH ARTICLE OPEN ACCESS

Autonomous AI Agent Testing and Safety Validation Framework

A Comprehensive Enterprise Framework for Functional, Behavioral, Security, Safety, Tool-Use, Delegation, and Human-Oversight Validation of Autonomous Agents

Sanjeeve Kumar Gajadi*
*(HCL Technologies, Bengaluru, Karnataka, India)
Email: sanjeevekumar.g@hcltech.com

****

Abstract:
Autonomous AI agents differ from conventional software because their behavior emerges from models, prompts, memory, retrieved context, tool availability, policies, and interactions with other agents. Traditional unit testing cannot fully validate systems that plan, adapt, delegate, and act in dynamic environments. This paper proposes an Autonomous AI Agent Testing and Safety Validation Framework that combines deterministic tests, scenario simulation, adversarial testing, tool-use validation, policy compliance, delegation controls, recovery testing, human oversight, and production readiness gates. The framework defines test dimensions for task completion, planning quality, authorization, robustness, security, hallucination, unsafe action, prompt injection, data handling, cost, latency, and reversibility. It introduces sandbox environments, golden scenarios, red-team cases, stress tests, failure injection, regression suites, and risk-tier-based acceptance criteria. The proposed approach enables enterprises to validate autonomous agents more systematically before deployment and throughout their operational lifecycle.

Keywords — Autonomous AI Agents, Agent Testing, AI Safety, Agentic AI, Red Teaming, Tool Validation, AI Evaluation, Human Oversight, Security Testing, Enterprise AI

****

I. INTRODUCTION

Autonomous agents can produce different execution paths for similar inputs and may interact with systems beyond the model itself.

Testing must therefore evaluate end-to-end behavior, not only generated text.

II. TESTING OBJECTIVES

Testing should establish whether agents complete tasks correctly, remain within authority, handle uncertainty safely, recover from failure, and escalate appropriately.

Acceptance criteria should reflect use-case risk and business impact.

III. TEST ENVIRONMENT

Agents should be tested in isolated sandboxes with representative data, simulated tools, restricted credentials, and controlled external dependencies.

Test environments should support replay and deterministic evidence capture where possible.

IV. FUNCTIONAL AND WORKFLOW TESTING

Functional tests verify task completion, workflow sequencing, state handling, API interaction, and business-rule compliance.

Golden scenarios provide expected outcomes for common and critical workflows.

V. PLANNING AND REASONING VALIDATION

Validation should examine whether plans are relevant, bounded, efficient, and consistent with policy.

Excessive loops, unnecessary delegation, or inappropriate tool selection should be treated as failures even if the final output appears correct.

VI. SECURITY AND ADVERSARIAL TESTING

Tests should include prompt injection, malicious retrieved content, credential misuse, data exfiltration, privilege escalation, and unsafe tool invocation.

 ©IJSRED: All Rights are Reserved

Red-team scenarios should evaluate whether controls remain effective under adversarial input.

VII. DELEGATION AND MULTI-AGENT TESTING

Multi-agent tests should validate scope propagation, identity boundaries, conflict resolution, termination, and prevention of unauthorized privilege inheritance.

Delegation chains should remain auditable and bounded.

VIII. SAFETY AND HUMAN OVERSIGHT

High-impact actions should be tested for approval gates, reversibility, uncertainty handling, and escalation to humans.

Human reviewers should receive sufficient context to make informed decisions.

IX. FAILURE INJECTION AND RECOVERY

Testing should simulate model outages, tool failures, stale data, authorization denial, network latency, and partial execution.

Agents should fail safely, avoid repeated harmful actions, and use approved recovery paths.

X. PERFORMANCE AND COST TESTING

Validation should measure latency, token use, tool-call volume, task duration, retry behavior, and cost per successful outcome.

Resource limits should be tested to prevent runaway autonomous consumption.

XI. REGRESSION AND RELEASE GATES

Changes to model, prompt, tools, memory, data, policies, or orchestration should trigger relevant regression suites.

Release gates should require minimum performance, safety, security, and evidence thresholds.

XII. CONCLUSION

Autonomous AI Agent Testing and Safety Validation Framework expands enterprise testing beyond conventional software QA.

By combining scenario, adversarial, policy, delegation, recovery, and human-oversight testing, organizations can deploy agents with stronger confidence and control.

ACKNOWLEDGMENT

The author acknowledges the contributions of enterprise architecture practitioners, AI governance specialists, standards bodies, security professionals, data leaders, and technology researchers whose published work informed the concepts synthesized in this paper.

REFERENCES

[1] National Institute of Standards and Technology, Artificial Intelligence Risk Management Framework (AI RMF 1.0), 2023.
[2] National Institute of Standards and Technology, AI RMF Generative Artificial Intelligence Profile, 2024.
[3] ISO/IEC 42001:2023, Information technology — Artificial intelligence — Management system.
[4] ISO 31000:2018, Risk management — Guidelines.
[5] ISO/IEC 27001:2022, Information security, cybersecurity and privacy protection — Information security management systems.
[6] The Open Group, TOGAF Standard, 10th Edition, 2022.
[7] ISACA, COBIT 2019 Framework: Governance and Management Objectives, 2019.
[8] OWASP Foundation, OWASP Top 10 for Large Language Model Applications.
[9] MITRE, ATLAS: Adversarial Threat Landscape for Artificial-Intelligence Systems.
[10] OECD, Recommendation of the Council on Artificial Intelligence.
[11] SAP SE, SAP Business Technology Platform Documentation.
[12] SAP SE, SAP Business Data Cloud and SAP Datasphere Documentation.

 ©IJSRED: All Rights are Reserved