



A Metric-Driven Framework for Evaluating Large Language Models in Software Testing: Insights from Industry Experts

V. I. T. Perera
(MS24016520)

A THESIS
SUBMITTED TO
SRI LANKA INSTITUTE OF INFORMATION TECHNOLOGY
IN PARTIAL FULFILMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE IN INFORMATION SYSTEMS

December 2025

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.



Prof. Dilshan De Silva

Approved for MSc. Research Project:



MSc in IS Programme Co-ordinator, SLIIT

Approved for MSc:

Head of Graduate Studies, FoC, SLIIT

DECLARATION

This is to certify that the work is entirely my own and not of any other person, unless explicitly acknowledged (including citation of published and unpublished sources). The work has not previously been submitted in any form to the Sri Lanka Institute of Information Technology or to any other institution for assessment for any other purpose.

Sign:.....
Perera V. I. T.

Date:27/12/2025.....

ABSTRACT

A Metric-Driven Framework for Evaluating Large Language Models in Software Testing: Insights from Industry Experts

Isuru Perera

MSc. in Information Systems

Supervisor: Prof. Dilshan De Silva

December 2025

The integration of Large Language Models (LLMs) into software testing workflows has introduced new opportunities for automation, but also raised critical questions regarding the reliability, maintainability, and effectiveness of the generated test cases. This study addresses the lack of standardized evaluation practices by proposing and validating a comprehensive metric-driven framework: STEAM-LLM (Software Test Effectiveness Assessment Model for LLM-generated tests). Through a mixed-methods research design involving structured surveys and expert interviews, the framework identifies three core independent variables: Prompt Engineering Level, Human Intervention, and Input Specification Quality and evaluates their impact on Fault-Revealing Power and Maintainability & Stability. The inclusion of Edit Effectiveness as a mediator, along with contextual moderators such as Task Complexity, Developer Experience, and LLM Class, reflects the nuanced dynamics of LLM-assisted testing. Confounding variables including Baseline Project/Test Quality and Time Spent on Understanding the Task were also accounted for to ensure valid assessments. The framework was empirically validated and supported by metric thresholds (e.g., $\geq 80\%$ coverage, $\leq 10\%$ smell density), offering practical benchmarks for industry adoption. The findings demonstrate that structured prompts, strategic human oversight, and high-quality inputs are essential for producing reliable and maintainable LLM-generated test cases. This research contributes a theoretically robust and practically applicable model for evaluating AI-assisted software testing, laying the foundation for future experimentation, tooling, and integration into continuous development environments.

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to my supervisor, Prof. Dilshan De Silva, for his continuous support, insightful guidance, and constructive feedback throughout the course of this research at Sri Lanka Institute Information Technology. I am also deeply thankful to all the survey and interview participants whose valuable experiences and perspectives greatly enriched the development and validation of the evaluation framework. Finally, I extend my appreciation to everyone who informally supported me during this journey, offering encouragement, ideas, and feedback that helped shape this work.

TABLE OF CONTENTS

DECLARATION	ii
ABSTRACT.....	iii
ACKNOWLEDGEMENT	iv
TABLE OF CONTENTS.....	v
List of Figures	ix
List of Tables	x
Chapter 1 : Introduction	1
1.1 Motivation.....	1
1.2 Problem Statement	2
1.3 Research Gap	2
1.4 Research Objectives, Hypotheses & Research Questions.....	3
1.4.1 Research Objectives.....	3
1.4.2 Hypotheses & Research Questions	4
Chapter 2 : Background and Related Work	6
2.1 Conceptual Background.....	6
2.2 Existing Approaches or Techniques	6
2.2.1 Unit Test Generation.....	6
2.2.2 Property-Based and Mutation-Based Testing	7
2.2.3 Security and Penetration Testing	7
2.2.4 User Acceptance and Use Case Based Testing	8
2.2.5 Mobile Application Testing	8
2.2.6 Comprehensive Survey and Meta-Analysis	9
2.2.7 Evaluation Metrics and Practices	9
2.2.8 High-Level and Requirement-based Testing	10
2.2.9 Web Application Testing	11
2.2.10 Advanced Techniques and Hybrid Approaches.....	11
2.2.11 Domain-Specific, Practical Implications and Industry Adoption	12
2.2.12 Integration & Service-Oriented Testing.....	13
2.3 Comparative Analysis	13
2.4 Summary of Reviewed Literature	15
2.5 Identified Challenges and Research Gaps.....	17
2.6 Conceptual Framework.....	19
Chapter 3 : Methodology	21
3.1 Research Design.....	21
3.1.1 Type of Research	21
3.1.2 Justification for Research Design.....	21

3.2 Sampling Strategy	22
3.2.1 Target Population	22
3.2.2 Sampling Method & Justification	23
3.2.3 Sample Size & Justification	24
3.3 Development of the LLM Testing Evaluation Framework	25
3.3.1 Literature Review and Theoretical Foundation	25
3.3.2 Expert Consultation and Refinement	25
3.3.3 Survey-Based Validation of Variables	26
3.3.4 Semi-Structured Interviews	26
3.3.5 Finalization of Framework Structure	26
3.4 Data Collection Methods	27
3.4.1 Survey Questionnaire	27
3.4.2 Expert Interviews	27
3.5 Data Analysis Techniques	28
3.5.1 Quantitative Data Analysis	28
3.5.2 Qualitative Data Analysis	29
3.5.3 Integration of Quantitative and Qualitative Findings	30
3.6 Ethical Considerations	30
3.7 Resources and Budget	31
Chapter 4 : Significance and Contributions	32
4.1 Significance of the Study	32
4.1.1 Academic Significance	32
4.1.2 Practical Significance	32
4.1.3 Timeliness and Relevance	33
4.2 Contributions of the Study	33
4.2.1 Development of a Metric-Driven Evaluation Framework	33
4.2.2 Empirical Validation through Mixed-Methods Research	34
4.2.3 Practical Guidelines for Industry Adoption	34
4.2.4 Foundation for Future Research	34
Chapter 5 : Data Analysis and Findings	35
5.1 Overview of Data Collection and Participant Distribution	35
5.2 Validation of Dependent Outcomes and Thresholds	37
5.2.1 Importance of Dependent Variables	37
5.2.2 Proposed Thresholds for Evaluation Metrics	37
5.3 Validation of Hypotheses	38
5.3.1 Hypothesis 1 – Prompt Quality vs. Test Effectiveness	38
5.3.2 Hypothesis 2 – Human Intervention vs. Maintainability & Effectiveness (Mediated by Edit Effectiveness)	41

5.3.3 Hypothesis 3 – Input Specification Quality vs. Outcomes (Moderated by Experience).....	44
5.3.4 Hypothesis 4 – Impact of Complexity, Experience, and LLM Class.....	46
5.3.5 Hypothesis 5 – Impact of Time Spent & Baseline Project Quality.....	50
5.3.6 Spearman’s Rank Correlation	52
5.4 Summary of Findings.....	53
Chapter 6 : Evaluation Framework – STEAM-LLM.....	55
6.1 Overview and Design of the Framework	55
6.2 Independent Variables	57
6.2.1 Prompt Engineering Level	57
6.2.2 Human Intervention	58
6.2.3 Input Specification Quality	58
6.3 Dependent Variables.....	59
6.3.1 Fault-Revealing Power.....	59
6.3.2 Maintainability & Stability	60
6.4 Mediator Variable	61
6.4.1 Edit Effectiveness	61
6.5 Moderator Variables	61
6.5.1 Task Complexity	61
6.5.2 Developer Experience	62
6.5.3 LLM Class	62
6.6 Confounding Variables	63
6.6.1 Baseline Project/Test Quality.....	63
6.6.2 Time Spent on Task	63
6.7 Control Variables.....	63
6.7.1 Basic Data	63
6.7.2 Runtime Environment.....	64
6.7.3 Test Type	64
6.7.4 Summary of Framework Logic	64
Chapter 7 : Results and Discussions	66
7.1 Overview.....	66
7.2 Interpretation of Key Findings.....	66
7.3 Alignment with Existing Literature	67
7.4 Practical Implications.....	68
7.5 Limitations of the Study.....	69
7.6 Recommendations for Future Research	69
Chapter 8 : Conclusion and Future Directions	71
8.1 Conclusion	71

8.2 Future Directions	72
References.....	74

List of Figures

Figure 2.1 Distribution of Key Evaluation Metric Types Across Reviewed Studies	15
Figure 2.2 Conceptual Framework.....	19
Figure 5.1 Distribution of Job Roles	36
Figure 5.2 Distribution of Primary Testing Types	36
Figure 5.3 PQI vs Effectiveness.....	39
Figure 5.4 HITL vs MST	42
Figure 5.5 RIQI vs Outcome.....	44
Figure 5.6 Spearman’s Rank Correlation.....	52
Figure 6.1 STEAM-LLM Framework Visualization	56

List of Tables

Table 2.1 Comparative analysis of LLM-assisted software testing approaches	14
Table 5.1 Importance of Dependent Variables	37
Table 5.2 Proposed Thresholds per metric.....	38
Table 5.3 Overview of Hypothesis Validation.....	53
Table 5.4 Participant-Level Support Matrix	54