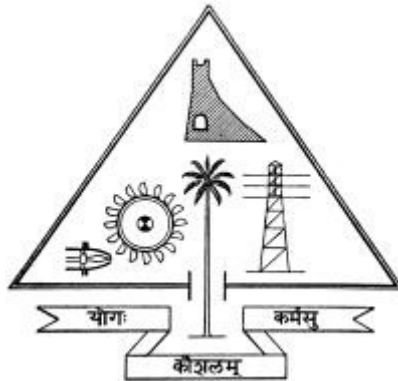


LLM-INTEGRATED SELENIUM FRAMEWORK FOR EXPLORATORY WEB TESTING

*Main Project Report submitted in partial fulfillment of the requirements for the
award of the degree of **Master of Computer Applications** of the **APJ Abdul
Kalam Technological University***

submitted by

**FATHIMA IRFANA NP
(TCR24MCA-2031)**



**DEPARTMENT OF COMPUTER APPLICATIONS
GOVERNMENT ENGINEERING COLLEGE
THRISSUR - 680009**

DECEMBER 2025

ABSTRACT

Modern web applications are increasingly dynamic and complex, making it difficult for traditional testing approaches to fully capture real user behavior. Users rarely follow predefined paths; instead, they interact unpredictably by exploring interfaces, providing unexpected inputs, and interrupting workflows. Conventional automated testing primarily validates known scenarios, while manual exploratory testing, although effective, is time-consuming and hard to scale. This project proposes an LLM-Integrated Selenium Framework for Exploratory Web Testing that aims to bridge this gap by simulating human-like exploratory behavior.

The framework combines Selenium’s browser automation capabilities with the reasoning abilities of large language models (LLMs). Selenium handles interaction with the web application by extracting visible elements, page states, and navigation outcomes. Based on this contextual information, the LLM dynamically reasons about possible user actions and selects meaningful next steps without relying on predefined scripts.

By making decisions at runtime, the system adapts naturally to modern single-page and dynamically rendered web applications. This intelligent, interaction-driven approach enhances defect discovery related to input validation, state management, error handling, and inconsistent UI behavior. Although challenges remain due to asynchronous updates and rapidly changing interfaces, the proposed framework demonstrates a flexible and scalable alternative to traditional scripted testing for effective exploratory web application testing.

REFERENCES

- [1] Wang, J., Huang, Y., Chen, C., Liu, Z., Wang, S., and Wang, Q. "Software Testing with Large Language Models: Survey, Landscape, and Vision." *IEEE Transactions on Software Engineering*, vol. 50, 2024, pp. 911–936.
- [2] Ayenew, H., and Wagaw, M. "Software Test Case Generation Using Natural Language Processing (NLP): A Systematic Literature Review." *Artificial Intelligence Evolution*, vol. 5, 2024, pp. 1–10.
- [3] Eidenbenz, R., Franke, C., Sivanthi, T., and Schoenborn, S. "Boosting Exploratory Testing of Industrial Automation Systems with AI." In *Proceedings of the 14th IEEE International Conference on Software Testing, Verification and Validation (ICST)*, Porto de Galinhas, Brazil, pp. 362–371, IEEE, 2021.
- [4] Job, M. A. "Automating and Optimizing Software Testing Using Artificial Intelligence Techniques." *International Journal of Advanced Computer Science and Applications*, vol. 12, 2021.
- [5] Sherifi, B., Slhoub, K., and Nembhard, F. "The Potential of LLMs in Automating Software Testing: From Generation to Reporting." *arXiv preprint arXiv:2501.00217*, 2025.
- [6] Mohapatra, P. S. "Artificial Intelligence-Driven Test Case Generation in Software Development." In *Intelligent Assurance: Artificial Intelligence-Powered Software Testing in the Modern Development Lifecycle*, 2025.