

existing tool. Hence, the precision of the web applications has improved by a proposed system. The proposed tool is to act as a testing tool to find faults from all LOC (line-of-code) in web applications. More advanced tools find faults during rendering process, so line coverage is less but this tool is act as testing tool, so line coverage is more compare to other tools.

6. References

- [1] Godefroid, Klarlund N, and Sen K, "DART: Directed Automated Random Testing," *Proc. ACM SIGPLAN Conf. Programming Language Design and Implementation*, pp. 213-223, (2005).
- [2] Sen K, Marinov D, and Agha G, "CUTE: A Concolic Unit Testing Engine for C," *Proc. ACM SIGSOFT Int'l Symp. Foundations of Software Eng.*, pp. 263-272, (2005).
- [3] Cadar C, Ganesh V, Pawlowski P M, Dill D L, and Engler D.R., "EXE: Automatically Generating Input of Death," *Proc. Conf. Computer and Comm. Security*, pp. 322-335, (2006).
- [4] Artzi S, Kie zun A, Dolby J, Tip F, Dig D, Paradkar, "Finding Bugs in Web Applications Using Dynamic Test Generation and Explicit-State Model Checking" *IEEE Trans. on Software Engg. Vol. 36, NO. 4, July/Aug (2010)*.
- [5] Shinde S K, Joshi S D., Kaushik D K., "Fault-Tolerant System for Dynamic Web Applications", *IJEDR, volume 3 issue 1, (2015)*, pp. 489-494.
- [6] Holzmann G J, "The Model Checker SPIN," *Software Eng.*, vol. 23, no. 5, pp. 279-295, (1997).
- [7] Benedikt M, Freire J, and Godefroid P, "VeriWeb: Automatically Testing Dynamic Web Sites," *Proc. Intel Conf. World Wide Web*, (2002).
- [8] Clause J and Orso A, "Penumbra: Automatically Identifying Failure-Relevant Inputs Using Dynamic Tainting," *Proc. Intel Symp. Software Testing and Analysis*, (2009).
- [9] Godefroid P, Levin M Y, and Molnar D, "Automated Whitebox Fuzz Testing," *Proc. Network Distributed Security Symp.*, pp. 151-166, (2008).
- [10] Cleve H and Zeller A, "Locating Causes of Program Failures," *Proc. Intel Conf. Software Eng.*, pp. 342-351, (2005).
- [11] Mishherghi G and Su Z, "HDD: Hierarchical Delta Debugging," *Proc. Intel Conf. Software Eng.*, pp. 142-151, (2006).
- [12] Csallner C, Tillmann N, and Smaragdakis Y, "DySy: Dynamic Symbolic Execution for Invariant Inference," *Proc. Intel Conf. Software Eng.*, pp. 281-290, 2008.
- [13] Zoufaly F, "Web Standards and Search Engine Optimization (SEO)—Does Google Care About the Quality of Your Markup?" (2008).
- [14] Shinde S K, Joshi S D, "Iterative Code Reviews System For Detecting And Correcting Faults From Software Code Documents" *IJARET Volume 5, Issue 11, November (2014)*, pp. 61-67.
- [15] Minamide Y, "Static Approximation of Dynamically Generated Web Pages," *Proc. Intel Conf. World Wide Web*, (2005).
- [16] More R. M., Shinde S K, "Framework to finding faults in Web Applications" (*GIS SCIENCE JOURNAL*) Volume 7 Issue 10, October (2020) pp. 218-221.
- [17] Shinde S K, Joshi S D, "Web Based Requirement Elicitation Tool" (*IJARCET*) Volume 4 Issue 9, September (2015) pp. 3719-3722.
- [18] Shinde S K, Joshi S D, "Schema Inference & Data Extraction from Templated Web Pages", *IEEE, International conference on Pervasive Computing (ICPC) 2015, Sinhgad College of Engineering, Pune, 10.1109/PERVASIVE.(2015).7087084, Jan.2015* pp. 1-6.