International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Automated vs. Manual Testing: A Productivity Comparison

Mohnish Neelapu

Category / Domain: QA Automation

Abstract:
Testing constitutes a significant module of software development process aimed at identifying defects within the software. It can be performed through both manual and automated methods. Automated testing involves utilizing an automation tool to evaluate the software system, and it was introduced to minimize the need for manual intervention. This paper discusses a comparative analysis of productivity among manual as well as automated testing models. The results indicate that automated testing yields superior productivity when compared to manual testing.

Keywords: Productivity, software testing, comparison, Automated and manual.

I. INTRODUCTION

In the last few years, software testing is gaining popularity and prominence in the software development field [1]. It is the methodology to render defect free software system. It provides a helping hand to detect errors, gaps or missing requirements in contrast to the actual requirements [2]. Software Testing must not be a separate stage in System development but must be an integral part throughout design development and maintenance stages [3]. Software Testing is generally employed in conjunction by terms confirmation & authentication "Running software in a test environment is known as software testing. Controlled environment, in an attempt to provide an answer to the following question: Does the software act as described. One of the methods to make system responsible is to thoroughly test the system. As software is a part of the system it needs a testing process too [4, 5]. Software testing not only makes the software product work well under all circumstances but also makes the product not work well under certain circumstances [6]. Various software types have various types of requirement. Aimed at instance, software in game is quite distinct as of bank software. Game user's requirement is different from bank user [7]. In all these situations when an organization produces or makes an investment in a software product it must guarantee that the software product will be acceptable to its end users, [8] its target customers, its buyers, as well as other people involved. Software testing is an attempt to make this assessment. In software testing, two modes exist namely manual as well as automatic. Manual creation of test cases and running them without any support from tools is referred to as manual testing [9]. Manual testing of software is done by a person who is sitting in front of a computer methodically browsing application screens, experimenting with different usage and input combinations, checking the outcome [10] against the predicted behavior and documenting their findings. Automation Testing refers to the process of utilizing an automation tool to run test suite. Objective of automation is toward minimizing quantity of test cases to be executed manually also not to eradicate manual testing completely. Few automation tools are: Winrunner, Loadrunner, JUnit, Silktest [11] etc. In this article

suggested a comparison of productivity between automatic and manual testing were explained. Also, the productivity like execution, speed, accuracy, initial setup expense, test repeatibility, scalability etc, were compared.

A. Objective

The primary goal of this study is toward contrast the productivity of automated along with manual testing. Further, testing parameters like execution, speed, accuracy, initial setup cost, test repeatability, scalability etc, were compared.

II. LITERATURE REVIEW

In 2021 Halani, K.R et al., 2021 [12] have performed both the types of testing on a website "Impressioncart.com" to compare and analyze the performance of both. It was clear that in the majority of automation technique cases performed better than manual technique then toward conclude that manual testing needs to be eliminated isn't reasonable. In order to test or verify some software application with precise tester need toward perform together methods because both have some merits and demerits. Testing is not merely restricted to identify the bug only but is much broader and must be implemented in project right from the beginning till retirement. Bezbaruah, A et al., [13] in 2020 explain the automation of tests performed manually on HIL test bench setup. Manual execution of tests has the drawback of taking longer compared to automated test execution in terms of test bench setup and test execution. A HIL manual test bench configuration includes hardware such as PCAN adapter, wiring harness, electronic/after-treatment control module (ECM/ACM) as well as various sensors along with actuators. Automation is achieved via NI Test Stand employing LUIS Box scripting as the simulator for the engine and ECM as well as LUIS specific wiring harness together by the hardware utilized in the physical test bench set up. In 2020 Singh, N et al., [14] identify and detail some of the situations which may illustrate differences in automated versus manual methods of penetration testing. There are certain situations where Finding weaknesses in online applications is better done by hand screening rather by automated scripts or vulnerability scanners applications. In certain other situations, the reverse might be the case. Automation tools and scripts have been utilized and tested to determine what might go wrong if attackers take advantage of such vulnerabilities. Also, some situations have been employed which decide if one method is superior to the other for discovering/detecting web application security problems. Lastly, the work finishes by giving results in terms of pros-and-cons of both methods, which it realizes after doing so.

A. Research gap

In the above existing works, the authors found that both automated Vs manual testing gives both advantages and disadvantages. They have used different web application scenarios to check the comparison of automated along with manual testing.

The structure of the paper is defined by: Section 1 explains the introduction, section 2 explains literature survey, section 3 explains proposed methodology, and section 4 explains result and discussion and conclusion part in section 5.

III. METHODOLOGY

Here the proposed methodology defines a productivity contrast between automated testing and manual testing. For it, some of the productivity criteria like execution, speed, precision, initial configuration

expense, reusability of test, expandability etc were investigated. Even some of the manual testing measurement like execution, speed, accuracy, initial setup cost, test repeatability, scalability etc.

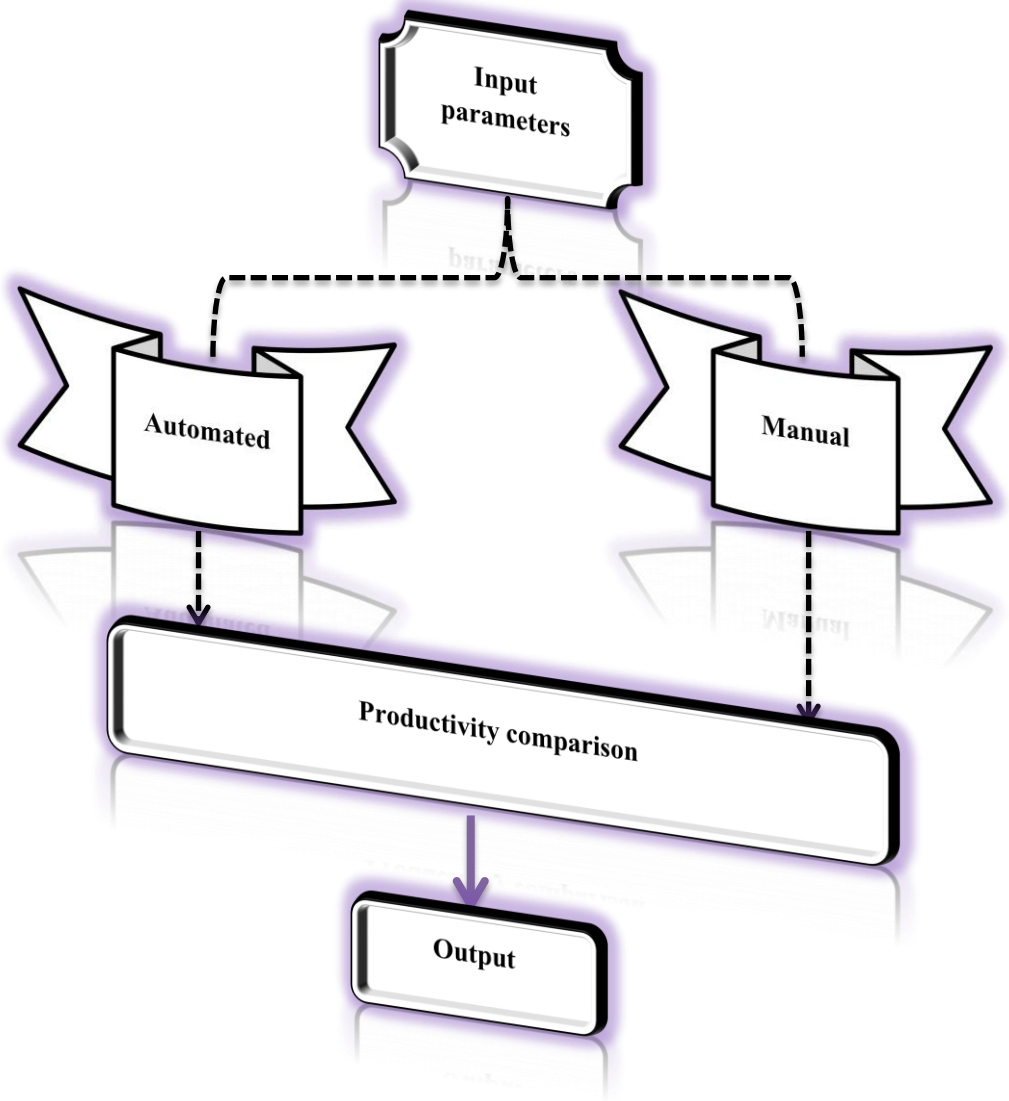

Fig. 1. proposed model.

A. Manual Testing

Manual testing is a method in which a test engineer physically develops also executes test cases to identify defects within software. This approach is considered one among the most rigorous and traditional methods of software testing. Manual testing can be a labor-intensive process that requires the [18] tester to possess a specific set of qualities, including patience, attentiveness, analytical thinking, creativity, open-mindedness, and proficiency. Conducting manual testing repeatedly can prove challenging, particularly for large software programs or ones with a lot of data. The proposed workflow model for productivity comparison is depicted in Fig. 1.

Working: Manual testing is somewhat interactive. Analysts and QA engineers must be heavily involved in every step of the process, from developing test cases to carrying out the tests themselves.

B. Automation Testing

Automating software testing entails the creation of test scripts utilizing scripting languages like Python, JavaScript, or Tcl (Tool Command Language) [19]. This allows for the execution of test cases by computers with limited human oversight and involvement. The processes of test additionally, design and development can be automated to cut expenses and human labor. Furthermore, automation software is capable of inputting test data into the system being evaluated, comparing anticipated outcomes with actual outcome, and producing comprehensive test reports. Test automation requires.

Working: Writing test scripts that automate test execution is known as automated testing. A collection of instructions to be followed on target platforms in order to verify a feature or anticipated result is called a test script.

C. Problems in manual testing

Time consuming also tedious: Because test cases are run via human resources therefore it is incredibly sluggish and tiresome.

Massive investment in human capital: Since test cases have to be run physically so there will be a need for more testers in manual testing.

Less dependable: Manual testing is less dependable since tests will not be executed with accuracy every time due to human mistakes.

Non-programmable: None of the programming is possible to develop complex tests which retrieve concealed info. Manual Testing may get tedious and thus error-prone [16].

D. Benefits of automation testing

Fast: It takes less time with manual testing.

Cost Effective: Test cases are run through the use of automation tool therefore fewer tester are needed in automation testing

Repeatable: Test case (record and replay) can be repeatedly executed using test tools [17].

Reusable: Test suits are reusable on varied software version.

Programmable: Testers are able to write complex tests which reveal underlying information.

Comprehensive: Testers are able to create test sets of tests that test all features in software application.

More reliable: Automation tests execute exactly same action every time they are executed.

Test Coverage: Broader test the application's feature coverage.

IV. RESULT ANALYSIS

In this section explains the comparison of automated Vs manual testing using charts and tables. The automation testing is more reliable without any human needs [20].

A. Comparison

A comparison between automated and manual testing has been conducted using a few criteria. Consequently, it was discovered that automated testing is very effective.

TABLE I

PRODUCTIVITY COMPARISON OF AUTOMATED VS MANUAL TESTING

Parameters	***Automated***	***Manual***
Execution speed	*Fast*	*Slow*
Accuracy	*Less human error with high accuracy*	*Human error and can vary.*
Recording of test cases	*High*	*low*
Test repeatability	*High*	*Low*
Scalability	*For large project this is highly scalable*	*Suited well for fewer projects*
Flexibility	*Based on script, flexible less*	*Highly flexible*
Programming knowledge	*Need*	*Non need*
Observation	*Fully automated*	*Need human observation*
time	*Speed processing time*	*Less processing time*
ROI	*Better ROI*	*Less ROI*
Execution	*Easy parallel execution*	*Hard parallel execution*
Need of skill	*For scripting and tools need knowledge*	*Suits fundamental knowledge of software*
Long-term cost	*Effective*	*Over time it become costly*
First setup cost	*High*	*low*
User experience testing	*Capability is limited*	*Excellent*
Developers feedbacks	*Continuous, rapid*	*Manual assessment based, slower*
reliability	*more*	*less*
Tool availability	*Need*	*Not need*

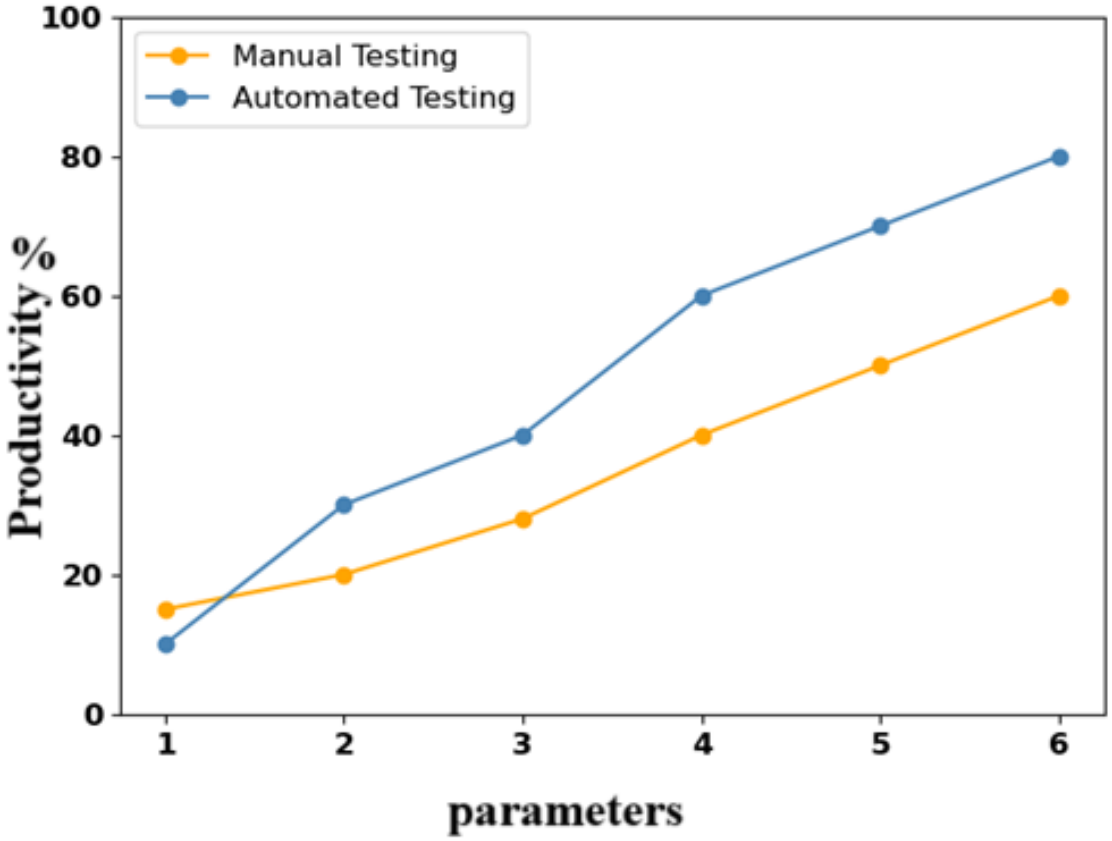

Fig. 2. Productivity comparison.

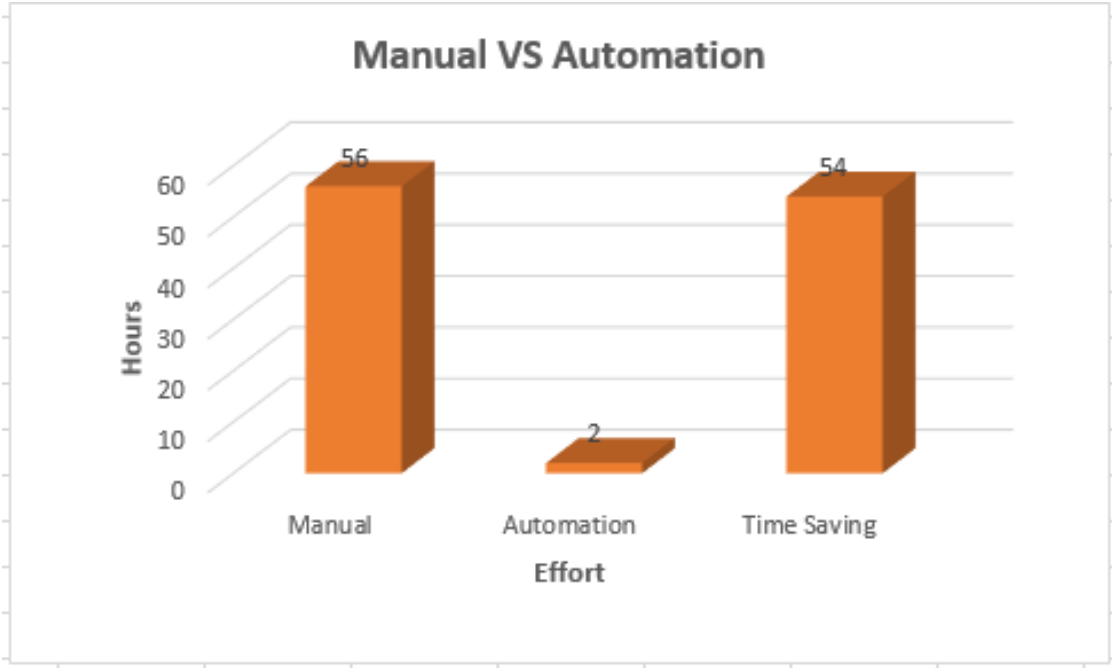

Fig. 3. Comparison of time.

Fig. 2 explains the productivity comparison of automated Vs manual test automation. X-axis represents Parameters of productivity and y-axis explains productivity % in both testing. Fig. 3 explains time

comparison of automated vs manual testing. Automation testing achieved less time taken. Fig. 4 explains the automation effort comparison for automatic and manual testing. Automation effort in automatic testing gives the high effort outcomes.

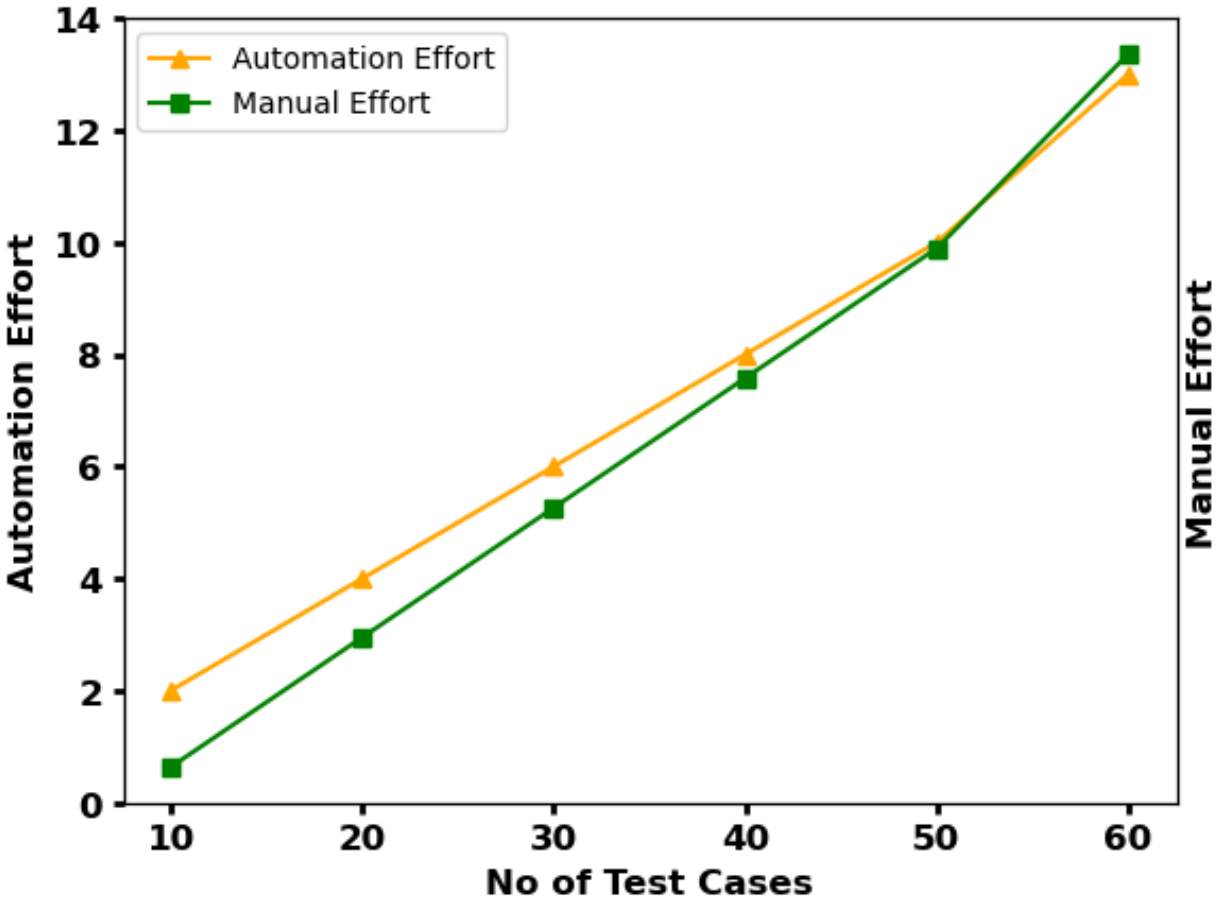

Fig 4: Analysis of automation effort

V. Conclusion

Manual testing is often characterized by its time-consuming and labor-intensive nature, necessitating significant human resource investment. Automation tools facilitate the recording of test suites, allowing for their subsequent replay as needed. Once a test suite is automated, it operates without the need for human involvement. Although the initial costs associated with automation testing are generally higher than those of manual testing, it is important to recognize that not all test cases can be automated. Therefore, it is essential to identify which test cases—whether manual or automated—will yield the greatest return on investment. This paper discussed some parameter productivity of both automated vs manual testing was analyzed. The automated test machine is the better choice to proceed in software testing. In future AI will implement for this comparison analysis using real data software.

Reference

1. Nama, P., Meka, N.H.S. and Pattanayak, N.S., 2021. Leveraging machine learning for intelligent test automation: Enhancing efficiency and accuracy in software testing. International Journal of Science and Research Archive, 3(01), pp.152-162.

2. R. Mukherjee, and K. S. Patnaik, "A survey on different approaches for software test case prioritization," *Journal of King Saud University-Computer and Information Sciences*, vol. 33, no. 9, pp. 1041-1054, 2021.
3. J. Shahid, M. K. Hameed, I. T. Javed, K. N. Qureshi, M. Ali, and N. Crespi, "A comparative study of web application security parameters: Current trends and future directions," *Applied Sciences*, vol. 12, no. 8, pp. 4077, 2022.
4. Fisher, M., Mascardi, V., Rozier, K.Y., Schlingloff, B.H., Winikoff, M. and Yorke-Smith, N., 2021. Towards a framework for certification of reliable autonomous systems. Autonomous Agents and Multi-Agent Systems, 35, pp.1-65.
5. X. Liu, Q. Zhou, J. Arulraj, and A. Orso, "Automatic detection of performance bugs in database systems using equivalent queries," *In Proceedings of the 44th International Conference on Software Engineering*, pp. 225-236, 2022, May.
6. Joshi, S. and Kumari, I., 2022, November. Analyses of software testing approaches. In 2022 International Interdisciplinary Humanitarian Conference for Sustainability (IIHC) (pp. 1276-1281). IEEE.
7. Salimon, M.G., Aliyu, O.A., Yusr, M.M. and Perumal, S., 2021. Smartphone banking usage in Nigeria: Gamification, technology acceptance and cultural factors empirical perspectives. The Electronic Journal of Information Systems in Developing Countries, 87(4), p.e12174.
8. Li, Z., Lu, S., Guo, D., Duan, N., Jannu, S., Jenks, G., Majumder, D., Green, J., Svyatkovskiy, A., Fu, S. and Sundaresan, N., 2022, November. Automating code review activities by large-scale pre-training. In Proceedings of the 30th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (pp. 1035-1047).
9. R. Mischke, K. Schaffert, D. Schneider, and A. Weinert, "Automated and manual testing in the development of the research software RCE,' *In International Conference on Computational Science. Cham: Springer International Publishing*, pp. 531-544, 2022, June.
10. Hashimoto, J., Ustun, T.S., Suzuki, M., Sugahara, S., Hasegawa, M. and Otani, K., 2021. Advanced grid integration test platform for increased distributed renewable energy penetration in smart grids. Ieee Access, 9, pp.34040-34053.
11. Roziere, B., Zhang, J.M., Charton, F., Harman, M., Synnaeve, G. and Lample, G., 2021. Leveraging automated unit tests for unsupervised code translation. arXiv preprint arXiv:2110.06773..
12. K. R. Halani, and R. Saxena, "Critical analysis of manual versus automation testing," *In 2021 International Conference on Computational Performance Evaluation (ComPE). IEEE*, pp. 132-135, 2021, December.
13. A. Bezbaruah, B. Pratap, and S. B. Hake, "Automation of tests and comparative analysis between manual and automated testing," In 2020 IEEE Students Conference on Engineering & Systems (SCES). IEEE, pp. 1-5, 2020, July.
14. N. Singh, V. Meherhomji, and B. R. Chandavarkar, "Automated versus manual approach of web application penetration testing," *In 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT). IEEE*, pp. 1-6, 2020, July.
15. A. S. Gadwal, and L. Prasad, "Comparative review of the literature of automated testing tools," *Researchgate*, vol. 10, pp. 13140, 2020.
16. Haas, M.A., Teare, H., Prictor, M., Ceregra, G., Vidgen, M.E., Bunker, D., Kaye, J. and Boughtwood, T., 2021. 'CTRL': an online, Dynamic Consent and participant engagement platform

working towards solving the complexities of consent in genomic research. European Journal of Human Genetics, 29(4), pp.687-698.

17. Hanif, H., Nasir, M.H.N.M., Ab Razak, M.F., Firdaus, A. and Anuar, N.B., 2021. The rise of software vulnerability: Taxonomy of software vulnerabilities detection and machine learning approaches. Journal of Network and Computer Applications, 179, p.103009.
18. Van Atteveldt, W., Van der Velden, M.A. and Boukes, M., 2021. The validity of sentiment analysis: Comparing manual annotation, crowd-coding, dictionary approaches, and machine learning algorithms. Communication Methods and Measures, 15(2), pp.121-140.
19. Paiva, J.C., Leal, J.P. and Figueira, Á., 2022. Automated assessment in computer science education: A state-of-the-art review. ACM Transactions on Computing Education (TOCE), 22(3), pp.1-40.
20. Ehsan, A., Abuhaliqa, M.A.M., Catal, C. and Mishra, D., 2022. RESTful API testing methodologies: Rationale, challenges, and solution directions. Applied Sciences, 12(9), p.4369.