STM JOURNALS

International Journal of
Electrical Machine Analysis and Design

Volume 1, Issue 2, 2023
July—December
DOI (Journal): 10.37591/IJEMAD

https://journals.stmjournals.com/iiemad

Review IJEMAD

Innovative Approaches for Exploring Software Testing Techniques for Electrical Equipments

Swapnil G. Tanwais[1,] *, Sushil Bakhtar[2]

Abstract

Software testing is critical to assuring the success and reliability of a software product or service, yet it is frequently regarded as uninteresting work when compared to design or development. Human factors influence the final software product's outcome, just like in any human-based activity. For software development companies, this means that finding practical ways to boost testers' motivation and job happiness is a major challenge. Our research focuses on how professional software testers might be motivated, we explore the policies and procedures that are developed and executed inside software development projects. Software testing is an unavoidable element of the Software Development Lifecycle and given its importance in the pre and post development processes, it should be addressed using improved and efficient procedures and techniques. It identifies and debugs system issues, malfunctions, and failures. Software development has evolved through various methodologies and strategies. The goal of testing is to maximize software efficiency. This process helps to guarantee that the software is built accurately and without errors by comparing the actual programme to the anticipated specifications. This method is used to enhance the usability, correctness, and efficiency of software in addition to identifying defects and faults in it.

Keywords: Software testing, software development, testing frameworks, automation, accuracy

INTRODUCTION

This process helps to guarantee that the software is built accurately and without errors by comparing the actual program to the anticipated specifications. This method is used to enhance the usability, correctness, and efficiency of software in addition to identifying defects and faults in it. Software testing assesses and improves software quality. Most organizations and testers suggest 40-50% of their resource (Time & Case) to testing. To achieve high accuracy, high levels of reliability, maintainability, availability, security, survivability, portability, [1–4] capability, efficiency, and integrity. There are many software testing techniques and strategies that are used for testing. Software testing involves reviewing findings against predefined requirements to ensure conformance and consistency. Various tests are conducted to identify both issues and potential outcomes of negative software testing. We check the system's correctness by ensuring that the user's specifications match their actual wishes. Early in the requirements assessment process, testing planning is initiated. It then methodically advances, undergoing continuous improvement throughout the software development process, until the programming stage is finished, and the test instances are started to run. Despite a large range of accessible tools, it is still an activity that requires a lot of human labor, i.e., a socio-technical rather than

***Author for Correspondence**
Swapnil G. Tanwais
E-mail: Swapniltanwais@gmail.com

[1]Final Year Student, Department of Electronics and Telecommunication, Prof Ram Meghe College of Engineering and Management, Amravati, Maharashtra, India
[2]Assistant Professor, Department of Electronics and Telecommunication, Prof Ram Meghe College of Engineering and Management, Amravati, Maharashtra, India

Received Date: March 06, 2024
Accepted Date: April 18, 2024
Published Date: April 25, 2024

Citation: Swapnil G. Tanwais, Sushil Bakhtar. Innovative Approaches for Exploring Software Testing Techniques for Electrical Equipments. International Journal of Electrical Machine Analysis and Design. 2023; 1(2): 1–7p.

© STM Journals 2023. All Rights Reserved

just technological activity, whose outcome will be highly dependent on the performance of the involved personnel. Unfortunately, research conducted in industry and among IT students indicates that many present and future software professionals regard testing as an unpleasant job. This could make it challenging to find and keep testers and low motivation can result in subpar testing and a failure to identify software bugs. Such issues are especially concerning at a time when the relative significance of testing to, say, coding is increasing because of more system integration projects and fewer green-field development initiatives, as well as a shift toward depending more on accessible components and services rather than writing code from scratch [5, 6]. Large-scale testing task automation may be able to alleviate some of the manpower shortages, but as said earlier, this is more of a long-term research objective than a short-term fix.

In the current digital era, electrical equipment from a variety of industries is rapidly integrating software. To improve usefulness, efficiency, and communication, software is essential to everything from consumer devices to industrial automation systems. However, strict testing methods designed to address the difficulties presented by software-hardware interactions are necessary to guarantee the dependability and safety of software-driven electrical equipment. This article explores the complexities of software testing methods with a focus on electrical equipment, emphasizing its utility, approaches, and uses.

Outsourcing and offshore will only work if a shortage of experienced testers in one company or country is offset by a surplus elsewhere; however, as previously said, the deficit could be global. Therefore, testing positions must be made more appealing. There is a dearth of research on human aspects in software testing. Bertolino offers a thorough analysis of a variety of software testing research challenges, addressing various methods with the question terms "why, how, how much, what, where, and when.". “Although there is substantial study on the motivation of IT people in general, as well as motivation in agile teams, there is, to our knowledge, a lack of research focusing especially on motivation in software testing.

Software Testing for Electrical Equipment: For a number of reasons, software testing is crucial to the electrical equipment development lifecycle.

- *Assurance of Safety:* Electrical equipment frequently works in settings where security is of utmost importance. Software flaws or malfunctions may result in dangerous events like electrical fires, equipment damage, or individual injuries. Thorough testing guarantees the equipment's dependability and integrity by assisting in the identification and mitigation of possible safety hazards prior to deployment. Functional Validation: Software regulates the operation modes, parameter settings, and communication protocols of electrical equipment, among other features. Testing confirms that the programme runs as planned, correctly deciphering user inputs, reacting to stimuli, and carrying out instructions to produce expected results.
- *Performance Optimisation:* There may be speed, accuracy, and efficiency requirements for electrical equipment. Testing assesses how well the programme performs in both typical and unusual scenarios, finding bottlenecks and enhancing responsiveness to reach performance goals.
- *Standards and Compliance:* Standards established by industries and regulatory bodies specify specifications for electrical equipment, such as cybersecurity, software dependability, and interoperability. Testing guarantees adherence to these standards, attesting to the equipment's conformity with industry and legal requirements for safety and quality.

PROPOSED SYSTEM

Software testing is the process of assessing a programme that is being tested for functioning. This procedure aids in confirming and validating the actual software with the anticipated requirements to guarantee that the product is correctly constructed and free of errors. This method is used to enhance the usability, correctness, and efficiency of software in addition to identifying defects and faults in it. Software testing's main objective is to find issues early in the SDLC (Software Development Life Cycle)

© STM Journals 2023. All Rights Reserved

in order to lower risks later. To quote the ANSI/IEEE 1059 standard, "Testing is the process of analysing a software item to detect the differences between existing and required conditions (that is, defects/errors/bugs) [7, 8] and to evaluate the features of the software item." Software testing can be done manually or through automation. While manual testing involves a tester to examine software manually, automation testing is performed by writing test scripts using any preferred programming language and executing them to evaluate the software against the requirements.

SYSTEM DEVELOPMENT

Block Diagram

The test bench employs virtual instrument technology, which is built on automated test systems. The software on the NI Lab View platform is divided into four levels with a hierarchical structure: test management, test procedures, instrument drivers, and I/O interface.

Figure 1 shows the test software function block diagram.

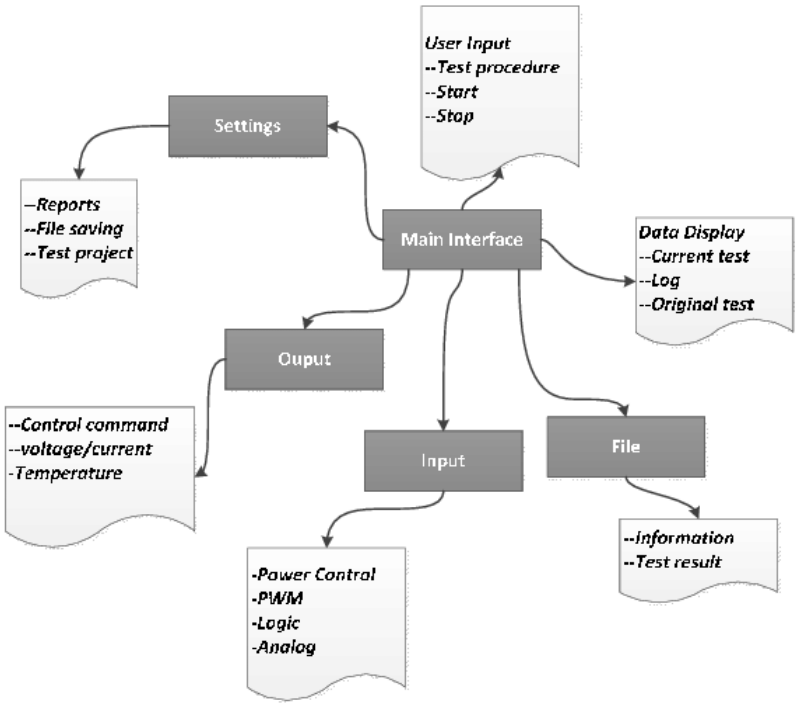

Figure 1. Block diagram of software testing.

Methods of Software Testing Electrical Equipment

Electrical equipment can benefit from a number of software testing methodologies, each of which focuses on a different facet of software functionality, dependability, and performance: Integration testing: System Testing:

Unit Testing

The first stage of testing, known as unit testing, is usually completed by the software engineers themselves. It is the process of making sure that each software component at the code level is operational and performs as planned. In a test-driven environment, developers often build and run tests before handing over the program or feature to the testing team. Although unit testing can be performed manually, automating the procedure can improve test coverage and reduce delivery cycles [9–10]. Debugging will be simpler as a result of unit testing since problems found early in the process will be easier to fix than those found later. Test Left is a solution that enables advanced testers and developers to shift left using the quickest test automation tool built into any IDE. Unit testing is the process of

© STM Journals 2023. All Rights Reserved

independently confirming that discrete software elements, like functions, modules, or algorithms, work as intended. Software-controlled components, such as motor controllers, sensor interfaces, or communication protocols, may be tested as part of unit testing for electrical equipment in order to verify their logic and behaviour.

Integration Testing

After each unit has been extensively tested, it is combined with others to form modules or components that are intended to execute specific functions or activities. Then, to make sure that all of the components of an application work as intended—that is, that the interactions between them are seamless—these are verified collectively through integration testing. Frequently, user scenarios—like opening files or logging into an application—are used to structure these tests. Integrated tests can be conducted by either developers or independent testers and are usually comprised of a combination of automated functional and manual tests. This type of testing assesses how several software modules or electrical equipment subsystems interact with one another. By identifying interface problems, data inconsistencies, and compatibility difficulties, this guarantees that linked components operate appropriately and interact fluidly.

System Testing

System testing is a type of black box testing where a finished, integrated system is assessed to make sure it meets the requirements. System testing evaluates the overall performance and functionality of the hardware, software, and external interfaces that make up an electrical equipment system. This extensive testing detects functional flaws, performance constraints, and interoperability problems while validating system behavior under realistic operational settings. Before the product is put into production, the software's functionality is tested from start to finish by a distinct testing team from the development team.

Acceptance Testing

Functional testing is followed by acceptance testing, which establishes if the finished product is prepared for release. It entails ensuring that the product adheres to all the original business objectives while also meeting the needs of the end users. As a result, the product must be tested both internally and externally. To do this, you must beta test it with end users and your quality assurance team. Beta testing is essential for gathering real input from future customers and addressing any remaining usability difficulties. This process verifies that the electrical equipment satisfies user expectations and needs. Acceptance testing involves end users or stakeholders assessing the equipment's usability, user happiness, and suitability for its intended application.

Usability Testing

Usability testing is a method of testing that evaluates an application's usability from the viewpoint of the user. It is often carried out in the course of acceptability or system testing. The purpose is to establish whether an application's visual design and aesthetics align with the planned workflow for tasks, such as signing in. Usability testing is an excellent technique for teams to assess if individual functions of the system are intuitive to use.

Regression Testing

Regression testing is a technique used to confirm that upgrades or modifications to software do not cause new bugs or regressions in already-existing functionality. Regression testing guarantees that software modifications do not impair the performance, safety, or dependability of electrical equipment.

RESULT

The final outcome of this study encourages the improvement of house mechanization, as the suggested arrangement suggests. With the help of this project, a computerized framework has been created that will enable us to operate household appliances including lights, fans, tube lights, air

© STM Journals 2023. All Rights Reserved

conditioners, bulbs, and so forth. One of the goals of this task is to achieve robotization and least effort venture. This document also includes information regarding the Arduino Uno, Bluetooth regulator, and hand-off module. Additionally, data regarding their work is provided. Along with the section of home computerization, its preferred position has also been investigated.

Some devices commonly used for testing purpose are presented in Figure 2

Figure 2. Testing equipment.

Identifying Defects in Software: The primary goal of software testing is to find and remove faults or bugs from the application. These flaws can range from minor glitches to serious faults, resulting in system crashes or erroneous data processing. Identifying defects in software is crucial for ensuring its quality and reliability. Software testing aims to confirm that the program operates as intended and complies with the specified specifications. In the case of a ticket booking app, functionality testing entails verifying that customers can easily reserve tickets, cancel bookings, and receive confirmation emails. To perform such functional testing, it is necessary to imitate user actions on the application to achieve more realistic results. Using Test sigma's AI-driven automated testing, you can simulate user interactions across many scenarios, ensuring the app's operation and delivering real-time feedback on its performance. You can use natural English to write test steps such as entering test data into a login form, pressing a button, and so on. Software testing techniques ensure that the application works properly across the various devices, platform, browsers. In various applications like ticket booking app. Ensure that it is compatible for various systems like smart phones, operating systems, and web browsers. Furthermore, integration testing ensures that the app works well with external services like payment gateways and third-party APIs.

Assessing Performance and Security: Performance and security are essential components of software testing. Performance testing assesses the application's ability to manage many concurrent users during peak booking periods. For a ticket booking app, this testing verifies that the app maintains optimal performance even under heavy user loads. Simultaneously, security testing seeks to identify flaws in data transfer, payment processing, and overall system security. By resolving potential speed and security issues, the app can retain user happiness while protecting critical user data.

Applications of Software Testing in Electrical Equipment

A variety of electrical equipment can benefit from the use of software testing techniques, such as:

© STM Journals 2023. All Rights Reserved

Industrial Automation Systems: Software is used for process control, monitoring, and data acquisition in programmable logic controllers (PLCs), human-machine interfaces (HMIs), and supervisory control and data acquisition (SCADA) systems. In industrial settings, testing guarantees the stability and dependability of these systems.

Power Distribution Systems: Software for energy management, control, and monitoring is integrated into smart metres, electrical grid infrastructure, and substations. Testing ensures that software-driven features in power distribution applications are accurate, responsive, and secure. Consumer electronics: Software is used by devices like wearable technology, smart appliances, and home automation systems to provide user interface, connectivity, and data processing. The interoperability, usability, and dependability of software features in consumer electronics products are validated through testing. Medical Devices: Software for data analysis, treatment control, and user interaction is included into electronic medical devices, diagnostic equipment, and patient monitoring systems. Testing makes sure that software-driven features in medical device applications are reliable, accurate, and safe.

CONCLUSION

To finish our survey, we would like to discover that software testing is a critical component of the Software Development Life Cycle. We can never call a product "perfect". Testing is never-ending. Testing merely indicates the presence of errors, not their absence. It is a time-consuming and demanding process; consequently, advanced approaches and creative methodologies are required to maintain software quality. This leads to the use of Automated Testing both before and throughout the testing process. The purpose of this paper is to define several software testing approaches in depth, as well as the importance of software testing and its goals and objectives. Software testing is frequently less formal and rigorous than it should be. To do software testing effectively and efficiently, people participating in the process must understand the fundamental concept, aims, and principles of software testing. Several software testing approaches are also covered, including security testing, white-box testing, black-box testing, and grey-box testing. Future testing-related activities will therefore rely significantly more on technology and use a model-based approach to automated testing and simulation, which will speed up the testing life cycle and provide effective quality assurance and optimal bug prevention. This results in the adoption of automated testing both before and throughout the testing procedure. The purpose of this paper is to provide a detailed description of several software testing approaches, as well as the necessity, goals, and objectives of software testing. Software testing is frequently not as rigorous and formal as it ought to be. The fundamental ideas, objectives, and principles of software testing must be understood by all parties engaged in the process for it to be carried out properly and efficiently.

REFERENCES

1. Atifi, M., Mamouni, A., & Marzak, A. (2017). A comparative study of software testing techniques. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)
2. N. Jenkins, “A Software Testing Primer”, 2018, pp.3–15.
3. P. Ron. Software testing. Vol. 2. Indianapolis: Sam’s, 2018.
4. M. S. Sharmila and E. Ramadevi. "Analysis of performance testing on web application." International Journal of Advanced Research in Computer and Communication Engineering, 2019.
5. S. Sampath and R. Bryce, Improving the effectiveness of Test Suite Reduction for User-Session-Based Testing of Web Applications, Elsevier Information and Software Technology Journal, 2018.
6. B. Pedersen and S. Manchester, Test Suite Prioritization by Cost-based Combinatorial Interaction Coverage International Journal of Systems Assurance Engineering and Management, SPRINGER, 201
7. S. Sprenkle et al., "Applying Concept Analysis to User-session-based Testing of Web Applications", IEEE Transactions on Software Engineering, Vol. 33, No. 10, 2007, pp. 643 – 658.
8. C. Michael, “Generating software test data by evolution, Software Engineering”, IEEE Transaction, Volume: 27, 2001.

© STM Journals 2023. All Rights Reserved

9. A. Memon, “A Uniform Representation of Hybrid Criteria for Regression Testing”, Transactions on Software Engineering (TSE), 2013.
10. M. K. Korucu, O. Kaplan, O. Buyuk, and M. K. Gullu, "An investigation of the usability of sound recognition for source separation of packaging wastes in reverse vending machines," Waste Management, Article vol. 56, pp. 46–52, Oct 2016.

© STM Journals 2023. All Rights Reserved