

# Testing Automation in Software Organizations

Nidhi Girish<sup>1\*</sup>, Prof Swetha S<sup>2</sup>

<sup>1</sup>Student, Dept. of Information Science and Engineering, RV College of Engineering, Bengaluru, India

<sup>2</sup>Professor, Dept. of Information Science and Engineering, RV College of Engineering, Bengaluru, India

\*\*\*

**Abstract** – All products that are released to the world are first tested to ensure that they satisfy the requirements they were built for. This testing previously used to take up as much, if not more time since it was done manually. However, with the introduction of automation in testing, organizations were able to drastically reduce the effort and time put into executing the tests. Testing can be accomplished automatically by using an application or developing a framework for the same. This paper looks into the testing process in general, and the current popular methods for testing automation practiced in software organizations.

**Key Words:** Software testing, Automation, Test case, Results, Test Script, Selenium

## 1. INTRODUCTION

Testing is an integral part of every software development cycle. It allows for developers and the organization to verify whether the application developed satisfies the demands of the client, and if it includes all the listed functionalities. However, it has been seen that majority of the test cases require Quality assurance analysts to spend extended amounts of time paying attention to the system and re-run tests in case of a minor manual error. There are several such discrepancies that need to be handled and pose unnecessary troubles in completing the testing process.

Automation helps improve the many issues pertaining to manual testing. It is the method of using applications or languages such as Python or JAVA to test systems and their features. This paper focuses on discussing the popular methods of automation testing in organizations, including both existing applications and developing languages. It aims to allow help readers get a better understanding of the subject, and make more informed decisions in testing.

The paper is arranged as follows. In section 2 the overview of Automated Testing is described. In section 3 the comparison between manual testing and Automation testing is given. In section 4, Software Automation Testing methods are elaborated. Section 5 of the paper is the conclusion of the Study Paper.

In the following subsection, the basic terms related to automation testing is explained. These are essential in understanding the next sections of the paper and get a clear picture of automation in general.

### 1.1 Basic Terms

**Testing:** Testing, according to the ANSI/IEEE is the process of analyzing a software product to identify the differences between existing and expected conditions (defects/errors/bugs) and to verify the features of the application or other software item. The purpose of testing is verification, error detection and validation in order to find problems and the purpose of finding those problems is to get them fixed. [1]

**Manual Testing:** This is the process of testing software manually for defects, that is, employing one or many testers to implement necessary commands on the system to verify each functionality promised by the product. It is known to be time consuming, and prone to human errors. Logging of the testing process becomes tedious as it is an extra work to be completed and cannot be done in parallel efficiently.

**Test Case:** A test case is the sequence of instructions and steps followed to confirm whether a specific functionality is as required. According to IEEE a test case is defined as “A set of input values, execution preconditions, expected results and execution post conditions, developed for a particular objective or test condition, such as to exercise a particular program path or to verify compliance with a specific requirement.” [2]

**Test script:** It is a set of instructions, developed using a programming or scripting language that is run on a system in order to test and verify that the system's performance is as expected. Test scripts are building blocks of automation testing.

## 2. OVERVIEW OF AUTOMATED TESTING

### 2.1 Automation Testing

Software test automation is the process by which the steps of manual test cases are automated using an automation tool or utility to shorten the test life cycle in time. When an error is detected, the objective of software testing is deemed to succeed. Software testing has its three main purpose i.e. Computer Quality Improvement (SQI), System Reliability Estimation (SRE), Verification and Validation (V&V).

### 2.2 Stages of Test Automation

Depending on the organization, the testing process is defined and customized based on their needs. However, the basic steps taken to automate test cases is usually the same. They are as follows:

- **Test automation Feasibility:** This step is to analyze the situation and deem if necessary to convert manual testing to automated testing based on many factors including time lost in keeping track of manual steps to be taken as well as how many errors occur.
- **Test automation Design:** Once decided to automate, a basic design layout needs to be established, listing the various libraries, tools and method of testing automation to be used. This template will help in the development phase to be uniform.
- **Test automation Development:** The test engineers, according to the framework described in the design phase, will begin the development of test scripts for each test case present.
- **Test automation Deployment:** once developed, the test scripts are run vigorously against testing systems to ensure they are successful in identifying errors, defects and lack of functionality.
- **Test automation Execution:** Once verified of success, all the test scripts are released to the customers as patches and their systems are tested regularly using these tests for proper functionality.

- **Test automation Maintenance:** In case of fails or improvements made into the systems, existing test scripts need to be upgraded to accommodate the new additions or fix errors and so on.

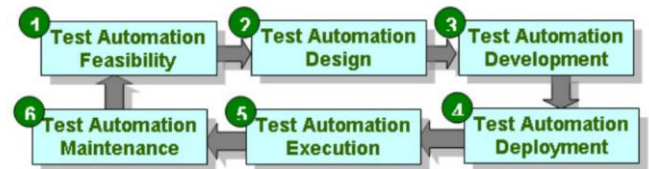


Fig 1: Testing Stages [3]

## 3. COMPARISON BETWEEN AUTOMATION AND MANUAL TESTING

The following are the comparative differences between automation testing and manual testing:

- Fast application development can be achieved by reducing testing time in automation compared to the slow process of manual testing.
- Automation testing is repeatable in nature, allowing for multiple runs of a test in the same speed and flow.
- Since it is repeatable, Automation testing is also reliable to give the exact same results every time.
- The scripts developed during automation are reusable and also programmable.
- Automation provides for regression testing without including additional resources
- Improvement in productivity is observed in automation testing due to reduction in manual involvement.
- Detailed test logs are available because it can be integrated into the script to allow for simultaneous logging.

## 4. SOFTWARE AUTOMATION TESTING METHODS

There exist several popular methods and tools available in the market for automation testing. Some of them are described below:

### 3.1 Selenium

Selenium is an open source instrument. It's a comprehensive collection of tools that supports rapid test automation creation for web-based applications. Selenium supports cross-browser testing and selenium testing can be executed on multiple browsers. It enables scripting in a number of languages, such as Java, C #, PHP and Python. Statement of assertion in selenium provides an effective way to compare the expected and actual results.

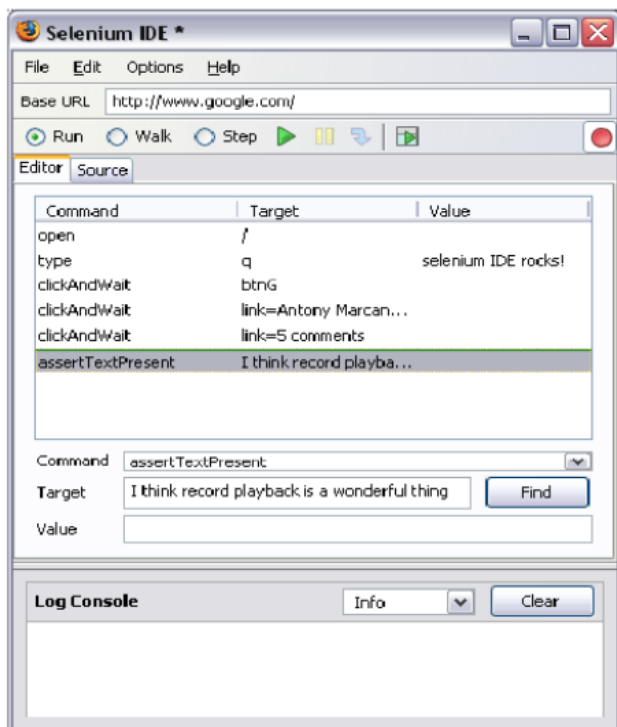


Figure 2: Selenium IDE window

### 3.2 Python

Unit Testing in Python is performed to find bugs early in the application's development stage when bugs are less frequent and less costly to repair.

A unit test is a structured code level test intended for testing a specific "unit" of functionality in Python. Unit test is an object-oriented frame focused on application features.

Python Unit Testing basically involves testing a specific module without having to access any reliant code. Developers can also use techniques such as stubs and mocks to split code into "units" and perform unit level testing on each piece.

### 3.3 JAVA

The Java programming language JUnit is a unit testing framework. JUnit was significant in the development of test-driven technology. Originating with SUnit, it is one member of a family of testing unit frameworks collectively known as xUnit.

At compile-time, JUnit is connected as a JAR; the framework resides under package junit.framework for JUnit 3.8 and above, and under package org.junit for JUnit 4 and later.

### 4. CONCLUSION

Test automation is used to reduce labor costs and other overheads. Automation results in decreased time spent on regression testing and offer companies an ability to increase the quality of their software products. Since automation technology can accomplish testing faster than human.

But automation testing efficiency depends on selecting the appropriate tool that is compatible with checklist, selecting test cases to automate. Also, cost is an important factor considered for tool selection.

All this is elaborated in this study paper to help professionals pick the best automation method that suits their needs.

### REFERENCES

- [1] Komal, 2013, REVIEW PAPER ON SOFTWARE TESTING, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) NCEAM – 2013 (Volume 1 – Issue 02)
- [2] Amaricai, Sabina; Constantinescu, Radu. Informatica Economica; Bucharest Vol. 18, Iss. 1, (2014).
- [3] Software Test Automation Abdul Rauf EMa\*, E.Madhusudhana Reddyb aResearch and Development Centre ,Bharathiyar University,Coimbatore-641014
- [4] Software Testing Research and Practice ISTI-CNR Italy
- [5] Achievements for software testing Mark Harman, Yue Jia and Yuanyuan ZhangUniversity College London, CREST Centre, London, UK

[6] Phil McMinn. Search-based software test data generation: A survey. *Software Testing, Verification and Reliability*, 14(2):105–156, June 2004

[7] Benny Godlin and Ofer Strichman. Regression verification: proving the equivalence of similar programs. *Software Testing, Verification and Reliability*, 23(3):241–258, 2013.

[8] C. Kaner, J. Bach, and B. Pettichord, *Lessons Learned in Software Testing*, John Wiley & Sons, 2002

[9] D. Graham, M. Fewster, *Experiences of Test Automation: Case Studies of Software Test Automation*

[10] LaShanda Dukes, Xiaohong Yuan, Francis Akowuah, “A case study on web application security testing with tools and manual testing”, IEEE 2013