

Generic Test Automation and Keyword Driven Approach

Bhagyashree Bhondokar¹, Pooja Ranawade², Snehal Jadhav³, Mayuri Vibhute⁴

¹bhagyashree61093@gmail.com, ²poojaranawade@gmail.com, ³snehal.snehal4@gmail.com, ⁴mayurivibhute@gmail.com

Dept. of Computer Engineering
MAEER'S MIT College of Engineering, Pune
Savitribai Phule Pune University

Abstract- In software development, testing plays an important role. Hence, it is necessary to have knowledge about testing. This paper provides an idea of testing and test automation. Simplifying testing efforts is the main objective of test automation. We have presented a process of Test Automation using keyword driven approach. The input in the form of natural language is converted to machine readable code snippet and that will be executed under framework to produce reports. The technique is based upon looking for keywords that describes actions on the target and calling functions associated with those. Using this framework, we can improve reusability of automated test. The effectiveness and efficiency of testing can be increased with the help of Automating test automation.

Testing improves the quality of

Keywords - Test Automation, Keyword, Framework, Manual Testing.

1. Introduction

1.1 Software Testing

Software testing is used to identify the accuracy, reliability & quality of developed software. It is the process of evaluating a system or system component by manual or automated means to verify that it satisfies specified requirements. Following are the concepts related to testing:

- 1) Test Data: For testing some feature of the software you need to enter some data as input. Any such data which is used in tests are known as test data.
- 2) Test Case: Test case is a smallest unit of testing, a snippet containing set of inputs, test environment, execution preconditions and expected output. Inputs are the specific values, tables, database, or may be file names.
- 3) Test Suite: Test suite is common term used for the collection of test cases. It is a container of the set of tests that helps tester in executing and reporting the test execution status.
- 4) Test Plan: Test plan narrates the whole strategy that team will follow, for testing the final implementation. The task of test plan is to regulate all testing activities. It includes: What to test? What strategy/method will be used for testing? Who will test the software? When to test the software? What risks are present?

1.2 Functions of Testing

Testing demonstrates that software function appear to be working accordingly to specifications. It makes software defect free so user can easily access software and makes better use of that software to carry out operations.

software so maintenance cost is reduced to great extent. Testing improves reliability and efficiency of the software.

1.3 Testing Approaches

To choose Test Approaches is one of major task to perform before actual testing starts. It is a step in test planning where the tester plans and documents how to go about testing. Considering various factors like risks, skills of the testers, stakeholders, product, business, cost etc. In a broad view, the testing approaches can be divided into two major types manual testing and automated testing.

- 1) Manual Testing: This type of testing is performed completely by human testers i.e. without using any automated tools or scripts. While testing the software the tester performs the role of end user and tests the software for any unexpected behaviour. Testers follow the test plan to ensure the completeness of testing.
- 2) Automated Testing: This approach is also known as test automation includes writing scripts and using software tools for testing. In this method, testing team can decide to test some parts of the software using test automation and some by manual testing. Automation testing helps in reusing the test scripts, reducing time required for testing and decreasing human errors.

Table 1. Comparative study of Manual and Automated Testing.

Factors	Manual Testing	Automated Testing
Time	Testers have to write each and every step hence it is very time consuming and tedious work.	As the tests are performed by software tool the execution is fast
Efficiency	Testers tend to reproduce human errors.	Software tool does not cause human errors and ensure accuracy.
Cost	Huge investment needs to be done in human resources.	Initial set up cost is more but once done can be reused without human intervention reducing cost.
Training	It does not require training for generating scripts	Testers need to be trained to use software tool to generate error free test scripts

1.4 Merits of Automated Testing

1. It is faster.
After the initial time given to generate the test scripts, the execution of automated tests is much faster.
2. It is more reliable.
Once test scripts are written and added to test suite they cannot be forgotten whereas manual tester can forget to perform some specific tests. Also automated tests are more accurate than manual tests as they do not involve human errors.
3. It reduces human and technical risks.
If the developer team changes automated test scripts will help them reuse the previously developed tests and thus reduce the risks.
4. It is more powerful and flexible
Using manual testing we cannot create 100 virtual users at a time which can be done by using automated testing. Also the test scripts can be reused.[4]

2. Test Automation Framework

It is not possible to automate every test case in a software test plan. Hence testers must first decide which tests are to be automated. Generally the tests which can be reused or take longer time to perform are automated. Furthermore testing all GUI items, connections with database, validations etc. can also be efficiently automated.[8] Following factors are mainly considered while deciding to automate tests:

- 1) Products that needs performing the same tests again and again.
- 2) Requirements of product do not change frequently.

Automation can be done by using languages like Java, vbscript and automated software tools. There are a lot of tools available that help in test automation. Typically JUnit[6] Automation Framework can be used for unit testing. Selenium[2][7] is used for web application testing along with Selenium Web Driver[7]. These tools help us to create testing framework.

A Test Automation Framework can be loosely defined as a set of abstract concepts, processes, procedures and environment in which automated tests will be designed, created and executed. In addition, it includes the physical structures used for test creation and implementation, as well as the logical interactions among those components. Figure 1 shows a view of framework structure. It consist of selenium[2] module which is important for interrelation of framework and web application under test.

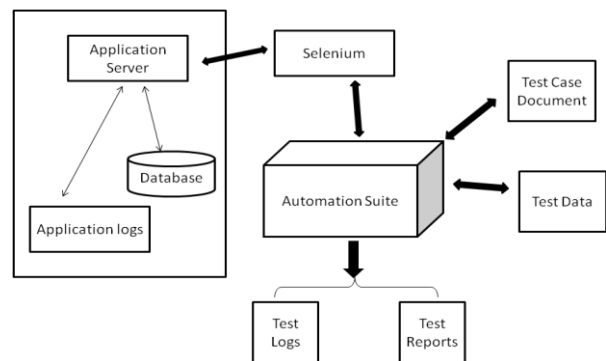


Figure 1. Overview of Framework Structure.

There are different types of frameworks :

1. Linear Framework–It is simplest & basic framework. Just a single program for sequential steps in a test script written. No modularity present.
2. Data driven Framework – This type is used to test the behaviour of an operation with variable set of data.
3. Keyword driven Framework– In this type, set of keywords are defined. Main program is based on the functions that are invoked with the help of keywords as parameters.

In Keyword driven Technique, it becomes possible to develop the scripts without having programming knowledge.

2.1 Keyword Driven Framework

The framework is said to be keyword driven when for a particular test case, first set of keywords are identified & then they are associated with a function (or action) with each of those keywords.[3] Figure 2 shows the generic flow of keyword driven approach.

It consists of Excel sheet and function libraries. Excel sheet consists of test case written manually in tabular format (as in UI element, Action, Target, Test data) [1] as shown in table 2. All the required logic to invoke the function has to be written in function libraries so that it will call different functions based on keywords after reading the excel sheets.

Label	Exists	BE 2015 Batch	
-------	--------	---------------	--

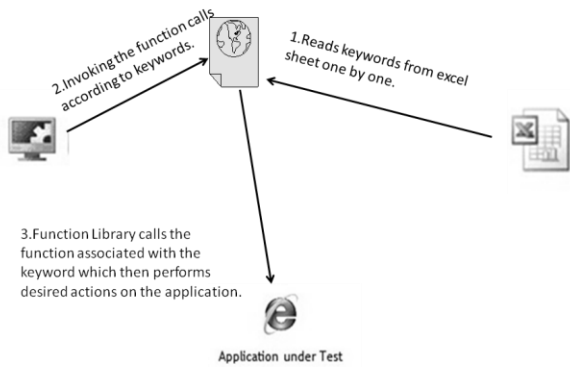


Figure 2. Work flow of keyword driven framework.

From the figure 2, you can see that the generic flow of a Keyword Driven Framework is something like this -

- 1) Test Script or Driven Script calls the main function library. Library contains code to identify the keywords.
- 2) Function library checks the keyword and searches for the first function that matches to the keyword.
- 3) Then it calls the function associated with this keyword.
- 4) All the required actions are then performed by the function on the application that is being tested.
- 5) Control returns to the main which then reads the next keyword (step 2). Steps 2 to 5 are repeated until the functions related to all keywords are called.

3. Automating Test Automation

Representing manual test case into the mechanically interpretable format such as representation in programming language (Java) is called as Test Automation.

In this paper , we are trying to produce automated test script from manually given input in the form of keyword table. Also this can be realized using a sequence of keywords which will automatically call their function calls. This keyword driven approach can be interpreted with the help of figure 2. We can see from table 2 except last row all others have certain task to do where as last step is for verification. It ensures that required state of browser is achieved.

Our task is to 1. Interpret the test cases written in natural language. 2. Create test scripts ready for execution. It is not easy to interpret natural language by computers so we will be focusing on basic parsing just to recognize keywords, actions and data. The test steps in keyword script will be despatched to our test application framework to generate test scripts automatically for final execution.[1]

Table 2. A script consisting of keywords and specified actions, their targets and any necessary data.

UI Type	ACTION	TARGET	TEST DATA
	Open		http://localhost:8080/main.jsp
Link	Select	Student Login	“Mitoce “
Text box	Enter	Username	“Mitoce”
Text box	Enter	Password	“BE2015”
Button	Click	Login	

Function calls will be generated from the keyword scripts those will produce test script. Test suite will contain various functions that are used while creating test script. Automation framework will then execute it and will produce report for test cases.

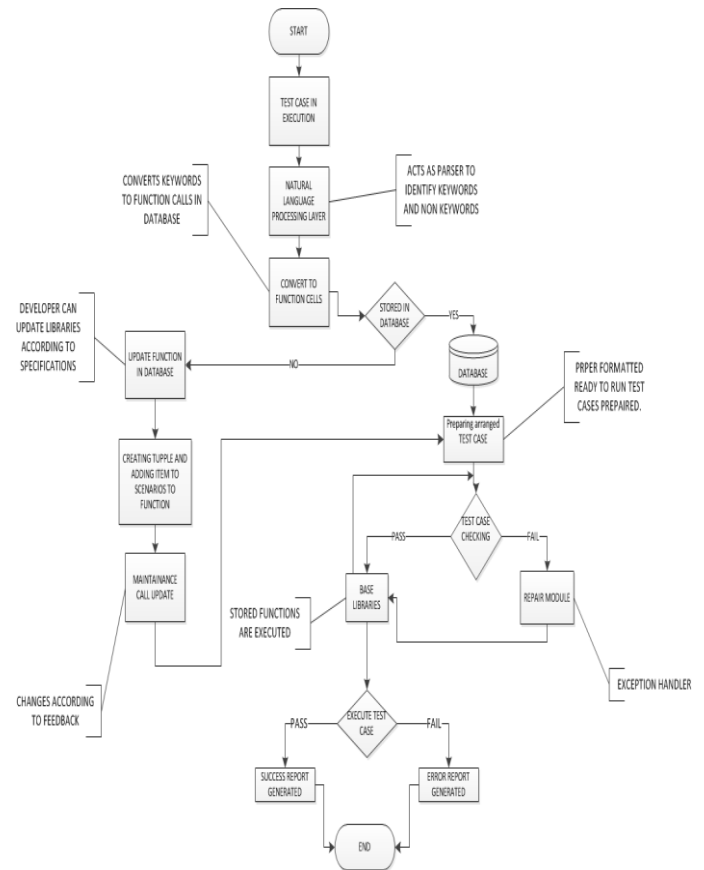


Figure 3 Overview of Automating Test Automation.

Our system will try to give complete process from test case generation to test script formation and their execution under automation framework generating report for the test.

This work will bridge the gap between novice programmer and testing domain. Tester will not have to learn the specific programming language to test the application. He will just fill the excel sheet in natural language to give keyword script to our framework system. After that our system will do the rest of work along with producing final report. If the test data given by the tester is not proper then only the system will fail to produce results. Otherwise, the tool will work without human intervention.

Tool will also contain the verification module to check whether given input is in the proper formatted excel sheet or not. Also we will add another module for repairing test scripts generated.

4. Related Work

There were certain conventional approaches for test automation like Record and Play technique. It is available with

many of the testing tools like RTF[12].It requires human to carry out test steps on the application under the test. But these kind of techniques needs human to understand test case and then execute it.

There are so many researches going on to have interpretation of natural language directly into programming language reducing human intervention[13]. This will further reduce effort for novice user while automating the test

automation. Related to our work, there are different techniques for automatically interpreting the natural language and form code snippet using keyword expressions[14], automating web applications using selenium[15].Also, technique to analyze use cases and models have been presented to detect defects and generate test cases[16]-[18].

Table 3 shows the literature survey of various papers and techniques presented in it for test automation.

Table 3. The Literature Survey.

Paper Referred	Methods Used	Advantage	Disadvantage
“Automating Test Automation” SureshThummalapenta, Saurabh Sinha, Nimit Singhania and Satish Chandra[1].	Data driven and keyword driven approach. Steps written in natural Language is input and the output is a sequence of procedure calls.	Steps written in natural Language is input and the output is a sequence of procedure calls.	No validations are used. Repair technique is not there.
“A Test Automation Framework Based on WEB” by Fei Wang Wencai Du,china[2].	designed an automatic software testing framework for web applications based on the Selenium and JMeter.	Load testing is performed precisely.	Unit testing is not performed. After completion ,software product is checked for load testing.
“Towards adaptive framework of keyword driven automation testing” Jingfan Tang Xiaohua Cao[3].	Designed an testing framework using XML .	Direct conversion of keyset variables to function calls(easy access)	Limited set of test drivers. Specialized inbuilt libraries not present.

5. Conclusion

This paper proposed the concept of testing and various testing approaches. It will help to understand test automation and its importance.

In this paper, we presented a technique to automate test automation using keyword driven framework. This is helpful to create test scripts and generate report automatically without human intervention using natural language input in the form of excel sheet keyword script.

6. References

[1] “Automating Test Automation” Suresh Thummalapenta, Saurabh Sinha, Nimit Singhania, and Satish Chandra† IBM Research – India (2012)
 [2] “A Test Automation Framework Based on WEB” ,Fei Wang , Wencai Du,china (2012)
 [3] “Towards adaptive framework of keyword driven

automation testing” Jingfan Tang (2008)
 [4] "Analytical Study on Manual vs. Automated Testing Using with Simplistic Cost Model" Prof.(Dr.)V. N.Maurya, Er.Rajender Kumar(2012)
 [5] “Software Testing” Second Edition By: Ron Patton, Pearson Education
 [6] “JUnit.” [Online]. Available: <http://junit.or>
 [7] “Selenium” “Selenium Web Driver”
 [Online] Available: <http://seleniumhq.org>
 [8] www.automatedtestinginstitute.com
 [9] www.automationrepository.com
 [10] www.softwaretestinghelp.com
 [11] www.m.softwaretestinggenius.com
 [12]“IBM rational functional tester.”
 [Online].Available:<http://www01.ibm.com/software/awdtools/ester/functional/>

[13] “Lowering the barriers to programming: A taxonomy of programming environments and languages for novice programmers” C. Kelleher and R. Pausch (2005).

[14] “Translating keyword commands into executable code,” G. Little and R. C. Miller(2006).

[15] “Co-scripter: Automating and sharing how-to knowledge in the enterprise,” G. Leshed, E. M. Haber, T. Matthews, and T. Lau(2009).

[16] “Application of linguistic techniques for use case analysis” A. Fantechi, S. Gnesi, G. Lami, and A. Maccari (2002)

[17] “A linguistic analysis engine for natural language use case description and its application to dependability analysis in industrial use cases” A. Sinha, A. M. Paradkar, P. Kumanan, and B. Boguraev, (2009).

[18] “Text2Test: Automated inspection of natural language use cases” A. Sinha, S. Sutton, and A. Paradkar(2010).