

# A Proposed Methodology for Automated E-Commerce Systems Testing

Khalid Alzubi  
Albalqa Applied University  
Salt, Jordan

Mai Alfawair  
Albalqa Applied University  
Salt, Jordan

## ABSTRACT

The goal of this paper is to explore the use of Keyword driven testing for automated testing. In Keyword driven testing, the functionality of the system under test is documented in a table as well as in step by- step instructions for each test. It involves creating tables which are assembled later on into test scripts. The Existing tools for this testing uses Html, Xml, Spreadsheet, and others.

## Keywords

Driven testing; test automation; test script; test results, scripting tool;

## 1. INTRODUCTION

Testing is an essential and most vital part of the software development life cycle (SDLC). The goal of the software testing is to find the errors or missing requirements. Testing is a process of evaluating the system or its components with the intent to establish whether or not it satisfies the specified requirements. Testing is divided into two kinds :manual and automation testing. Manual testing is carried out by a tester who tests the software manually. It requires a tester to play the role of an end user. Problems with manual testing include being a very time consuming process, not reusable, has scripting facility and some errors remain uncovered.

The automation testing means execution of test cases in an automated way without manual involvement. Test automation includes various activities like test generation, reporting the test execution results, and test management. All these test automation activities can take place on all the different test levels. These test levels are unit testing, integration testing, system testing, and acceptance testing. [2]

Benefits of Automation Testing include Saving time and money, improving test accuracy, ncreasing and increasing test coverage. Besides ,Complex testing is made easy and increases efficiency and team morale.

The test automation frameworks have evolved over time. They have evolved into three generations. [3] Figure 1 shows evolution of Test Automation. In the beginning, there was record and playback script creation. In this, there were only stand-alone test scripts. Later on came the Functional Decomposition. It consists of reusable functional; test modules.

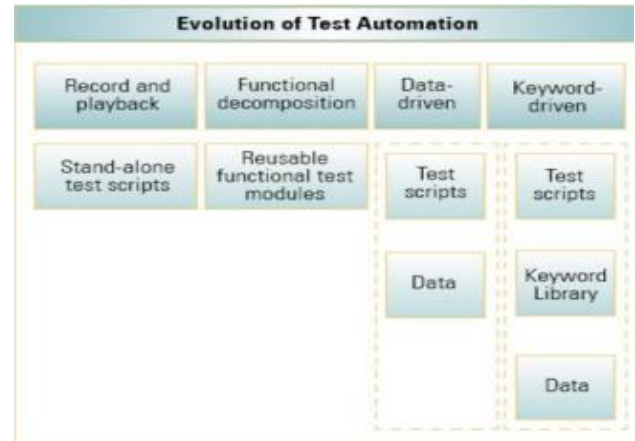


Figure 1. Evolution of Test Automation [7]

After that came data-driven testing. In this, test data is taken out of the scripts. This makes the test data variation easy and similar test cases can be created quickly. Today, keyword-driven testing is getting more popular. It is a technique that separates much of the programming work from the actual test steps so that the test steps can be developed earlier and can be maintained with only minor updates. It consists of test scripts, keyword library and data. Table 1 shows benefits and shortcomings of Automated Testing Approaches.

## 2. KEYWORD DRIVEN TESTING

Keyword-driven testing, also known as table-driven testing or action word based testing, shows Figure 2. and this methodology type of functional automation testing framework, is a software testing methodology suitable for both manual and automated testing, usually a spreadsheet, to define keywords or action words for each function that we would like to execute. This gives the chance to overcome the biggest limitation [1]

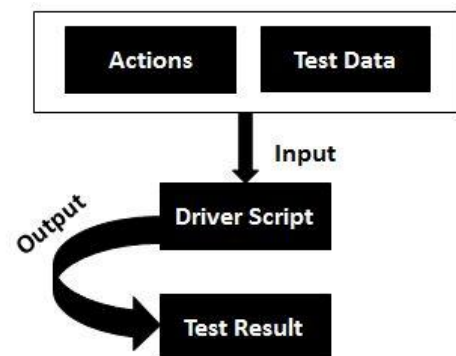


Figure 2 Keyword Driven Framework [1]

This methodology uses keywords (or action words) to symbolize a functionality to be tested, such as Enter Client. The keyword Enter Client is defined as the set of actions that must be executed to enter a new client in the database.

**Table i. Benefits and shortcomings of automated testing approaches [7]**

| Approach                 | How it works  | Benefits   | Shortcomings  |
|--------------------------|---|--|---|
| Record and Playback      | Users' action are captured, then played back on the application       | Ease of Scripting,   | Difficult to maintain test scripts,   |
| Functional decomposition | Re-usable   | Flexibility maintainability, reduces redundancy,                                       | meaning limited reusability, ease of maintenance  |
| Data-driven              | I/O data is maintained in external files                              | Size of the test pack is greatly reduced,  | Depends on technical expertise of test team   |
| Keyword-driven           | Robust, application independent re-usable keyword libraries are built | Ease of maintenance and highly scalable reduced dependence on application availability | Requires great deal of efforts and is time consuming, expertise in test tool scripting language required by framework development |

There are various advantages of using keyword driven testing techniques. These advantages are as follows:

- 1) Keywords that reflect the business can be chosen
- 2) Keyword re-use across multiple test cases
- 3) Not dependent on Tool / Language
- 4) The keyword list is robust to minor changes in the software.

**Table ii. Keyword-driven test data file**

| Type     | Keyword  | Operation | Parameter        |
|----------|----------|-----------|------------------|
| Function | Runapp   |           | http://ex.com    |
| item     | username | setValue  | Username=Ibrahim |

|      |          |          |                 |
|------|----------|----------|-----------------|
| item | password | setValue | Password=123456 |
| item | ok       | Click    |                 |

### 3. KEYWORDS

The test language comprises words (keywords) and parameters case is a sequence of steps that describe the behavior of future in an application. Test language uses the keyword to describe the steps and expected results. A test cases comprised of at least one keyword [4].there are three kinds of keyword items, utility function and user keyword.

Item is an action that performs a specific operation on a given GUI component such as set value "Ibrahim" in "User name" control, set value "123456" in "Password control result" field. When performing an operation on a GUI item, the following parameters should be specified: Name of GUI item, what operation to perform and the values. Table 3 shows Item operations.

Utility Function is a script that executes a certain functional operation that is hard or ineffective to implement as a Sequence. For example: Runapp, closeapp. Table 4 shows Utility Functions.

Sequence is a set of keywords that produces a business process, such as "Login". Sequence keyword is made by combining various items and utility function. It is recommended to collect frequently used functional processes such as login, addition of new records to the system as a sequence instead of implementing them as items in test cases. Table 4 shows Sequence keywords.

Parameters are additional information required in order to generate the test conditions. In most cases, parameters should be defined for the created keywords. For example, failed authentication by passing username with illegal password, number for mathematical calculation, etc.

Examples for sequence parameters: When the user wants to create a new user, the following syntax is used: create\_user (Ibrahim, 6/12/2000, Ibrahim.1306@yahoo.com) Some of the keywords may contain dozens of parameters. In order to simplify the test creation, all parameters should contain default values. The tester should be able to change each one of the default parameters according to the context of the test. For example, if the tester would like to create new user that is older the 100 years, only the birth date will be changed and all other parameters will remain the same. Obviously, a specific change will not affect the default parameters being used for other tests.

**Table iii. Item operation**

| Type | Keyword  | Operation | parameter        |
|------|----------|-----------|------------------|
| item | username | setValue  | Username=Ibrahim |
| item | password | setValue  | Password=123456  |

#### Base Requirements

A number of base requirements should be considered to success keyword driven .These include :-

It should separate test development and the automation.

The test cases should be clear in scope.

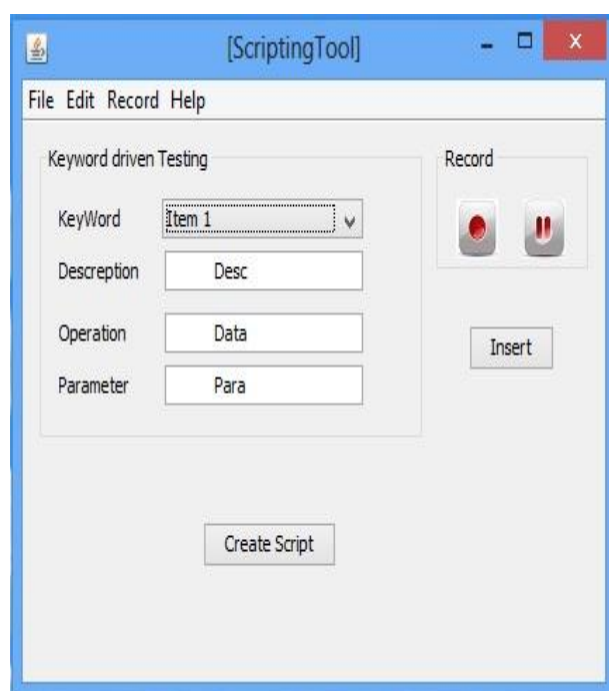
The test case should be in high level of accuracy.

**Table iv. Utility function**

| Type     | Keyword  | parameter     |
|----------|----------|---------------|
| Function | Runapp   | http://ex.com |
| Function | Closeapp | http://ex.com |

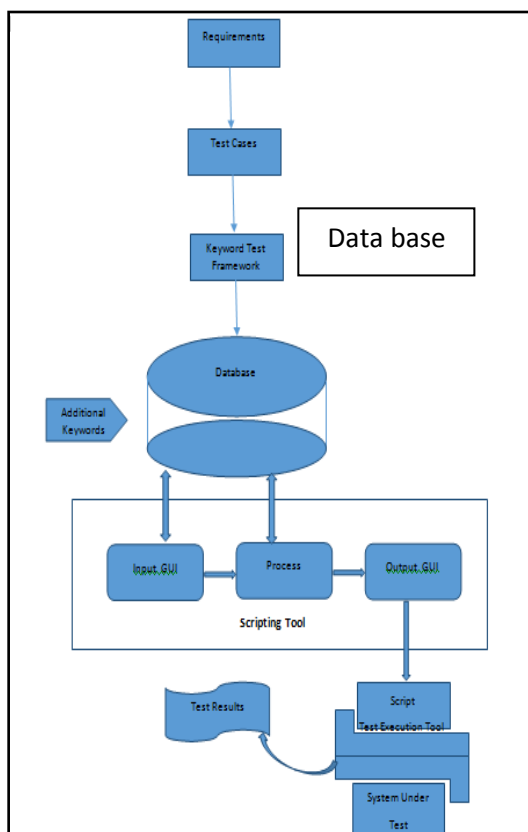
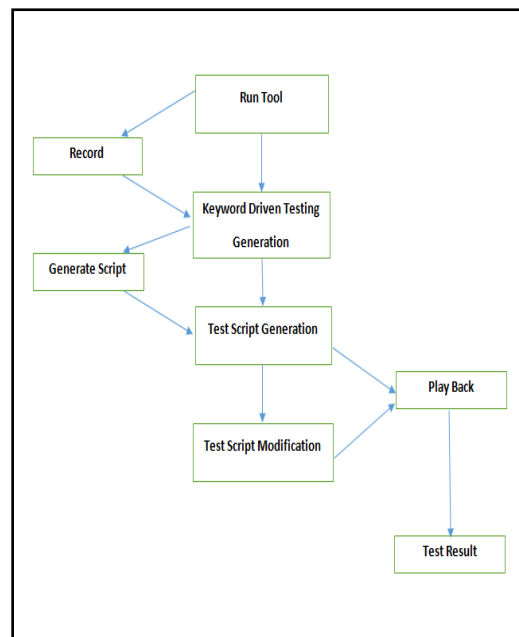
#### 4. PPOPOSED Methodology

One of the problems in the keyword driven test is that it needs an experience developer to solve this problem. For this, we suggested a scripting tool which consists of a GUI connected to the data base shown in figure 4, the data base contains test cases (tables) which are filled through the scripting tool.



**Figure 4 . Scripting Tool**

The scripting tool has two approaches to create a script record and play back which is used during the data entry as long as the data entry is confident of the accuracy of the data entry where a script is created through the button in GUI script tool. Once the test script is created, we need to play back in a version tool like UFT, QTP, etc. to get test result. We then check the result and if drastic modifications are needed, we must modify it and then run it again in the tool. Figure 5 shows the above proposed approach in a flow chart.



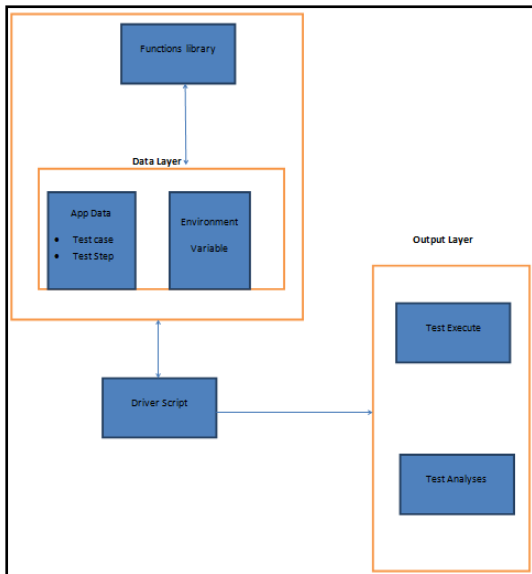


Figure 5. Proposed Methodology

## 5. CONCLUSION

In this paper, the different types of keywords, base requirement, methodology, and various keyword driven modules are investigated. These all are required to carry out a successful and efficient operation of Keyword driven testing. The process of the test automation should be carried out but it should not control the process. It should flow from the overall strategy, methodology, and architecture. Moreover, the existing tools available for this approach make use of the HTML, Xml, spreadsheets to maintain test cases in object repository which are not very scalable.

## 6. REFERENCES

[1] [http://www.tutorialspoint.com/software\\_testing\\_dictionary/keyword\\_driven\\_testing.htm](http://www.tutorialspoint.com/software_testing_dictionary/keyword_driven_testing.htm)

[2] Pekka Laukkanen, "Data-Driven and Keyword-Driven Test Automation Frameworks", Helsinki University of Technology, Department of Computer Science and

Engineering Software Business and Engineering Institute. 2007, pp. 1-102 .

- [3] Juha Rantanen, "Acceptance Test-Driven Development with Keyword-Driven Test Automation Framework in an Agile Software Project" Helsinki University of Technology, Department of Computer Science and Engineering, Software Business and Engineering Institute. 2007, pp. 1-102
- [4] Ayal Zylberman and Aviram Shotten, "Test Language: Introduction to Keyword Driven Testing". 2010, pp 1-7
- [5] Tommi Takala, Mika Maunumaa, and Mika Katara. "An Adapter Framework for Keyword-Driven Testing", Department of Software Systems, Tampere University of Technology, Finland. Ninth International Conference on Quality Software. 2009, pp. 201-210
- [6] [http://en.wikipedia.org/wiki/Keyword-driven\\_testing](http://en.wikipedia.org/wiki/Keyword-driven_testing)
- [7] Bharath Anand R., Harish Krishnankutty, kaushik Ramakrishnan, Venkatesh V.C., "Business Rules- Based Test Automation- A novel Approach for accelerated testing". 2007, pp. 1-12
- [8] Liu Xing, Li Yan, Cai Mian, Guo Ying, "The Testing and Evaluation
- [9] System for the Secure Operating System Based on the Mechanism of keyword-driven". Ninth International Conference on Information Assurance and security. 2009, pp. 471-474
- [10] [http://en.wikipedia.org/wiki/Test\\_automation](http://en.wikipedia.org/wiki/Test_automation)
- [11] <http://www.automatedqa.com/products/testcomplete/manager-overview/>
- [12] Bennett, "J.P. Introduction to Compiling Techniques – A First Course Using ANSI C", Lex and Yacc [M]. McGraw Hill Book Co, 1990.
- [13] Nancy S. Eickelmann, "An evaluation of software test environment architectures". International Conference on Software Engineering, 1996, pp. 353-364.