International Journal of Engineering Research and Technology. ISSN 0974-3154, Volume 13, Number 12 (2020), pp. 4662-4665

Automatic Testing for Web Application Using HP-ALM Tool

Rijwan Khan[1], Ayman Qahmash[2] and Mohammad Rashid Hussain[3]

[1]Professor, Department of Computer Science and Engineering, ABES Institute of Technology, Ghaziabad, Affiliated to AKTU, Lucknow, India. ORCID: 0000-0003-3354-3047.

[2]Assistnat Professor, Department of Information Systems, King Khalid University, Abha, Saudi Arabia.

[3]Assistnat Professor, Department of Information Systems, King Khalid University, Abha, Saudi Arabia.

Abstract

The primary objective of this paper is to showcase how automatic testing processes reduce the manual efforts and speedup the testing process. For the whole process of automation, we used a tool developed by HP. Authors customized automation framework leveraging the functionalities and benefits of standard Business Process Testing (BPT) framework provided by HP. HP-UFT (Unified Functional Testing) will be used as an automation development tool and integration of test flow will be created using HP-ALM tool.

Keywords: Business Process Testing; Unified Functional Testing, Genetic Algorithm, Software Testing, Automatic Testing.

I. INTRODUCTION

For this paper we created a web application "freight management" and tested it on both as manual and automated. We have name that application as Opti freight. Antifaith is an online freight management system which focuses on optimizing the freight movement. Opti freight application keeps on updating new features based on the customer's experience and requirements. For every new release the application goes through regression testing. As the application is expanding the manual scripts suite size is also increasing, so testing the application manually is becoming time taking and costly. Automation is the only solution to the above problem. This document covers business overview of the Opti freight application and the 9 applications that make up the Opti freight system. The test plan document along with test coverage, test approach and test management are explained latter in this paper.

A sample manual script has been enclosed in the test plan section of this document along with complete execution flow in ALM has also been described. This part contains manual execution effort which will later help us to derive ROI after automating the complete process automation features and requirements those are specific to the Opti freight suite of application are explained in details in the automation tool selection and automation fitment analysis sections along with fitment comparison matrix [1, 2, 3, 4]. Details of UFT tool with the functionalities it delivers have been enlightened in this paper.

II. TEST PLAN DOCUMENT

In the testing of web applications, the test plan documentation tracks the necessary information for defining the approach of testing. During the planning phase of a web application, the test plan document is created. Some parts of the test document plan can be shared with the customer for the approval of the input scenario of testing.

II.I Test Coverage

There are in total 9 applications under Opti freight and each application goes through changes in the requirements when there is a particular release. There are 68 manual test scripts in the regression suite which are executed in every release. All the 68 manual test scripts are given in table 1 for all the different applications which are running under our web app.

Table 1. Application Levels Test Coverage 1

S. No	Application Under test	Number of Manual Test Scripts
1	Customer Portal	24
2	Vendor Portal	3
3	Lean	5
4	LTL	3
5	Pricing	3
6	Orion	4
7	VRA	4
8	GMS	10
9	WUMQ	12
	TOTAL	68

II.II Test Management

HP - ALM is used as the Test Management tool. Test Plan creation, Manual Business Component creation, Test script execution and defect logging for the failed test cases are performed from the ALM test management tool. We changed the HP-ALM test management tool according to our requirement then applied testing.

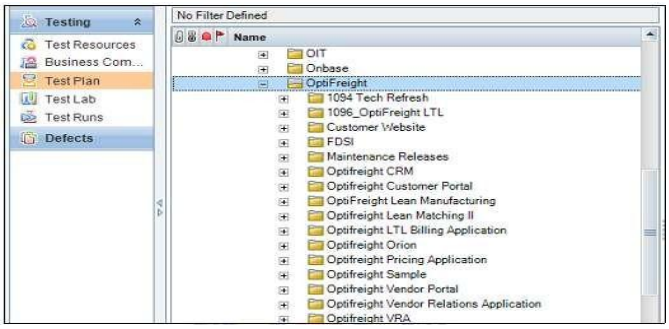

Fig. 1. Test plan view in ALM

In the ALM tool the testing criteria has different parts like test plan, test lab and test run etc. In this paper we define only two of the parts first is the test plan which is given in figure number 1 and the second is test lab which is given in figure number 2.

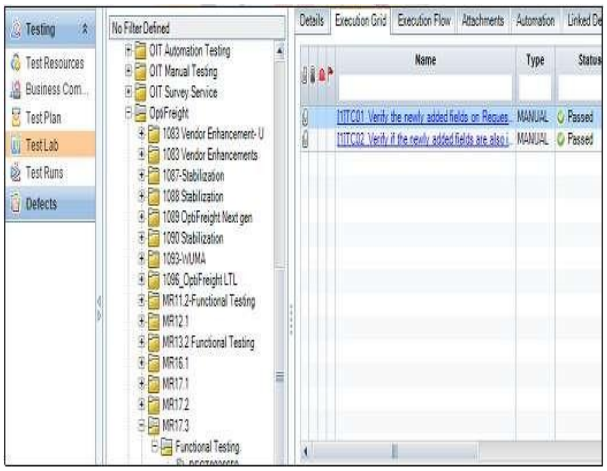

Fig. 2. Test lab view in ALM

III. MANUAL EXECUTION OF TEST CASES

ALM tool for manual execution of our web application is used and the manual script for ALM is already written. When we run manual test testing in ALM tool then it passed some test scripts and not run for the some which is given below in figure 3.

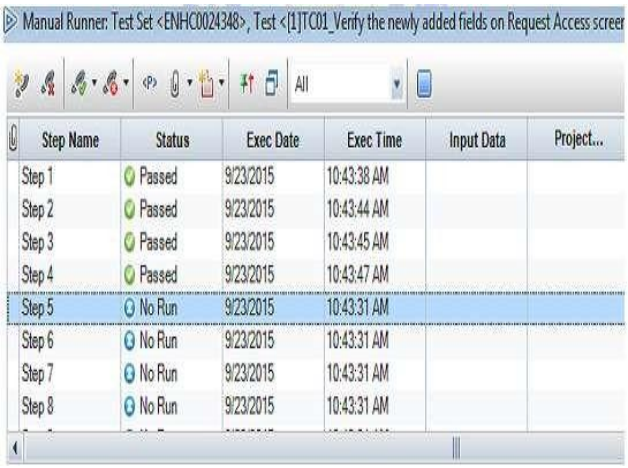

Fig 3. Manual Execution from ALM

IV. AUTOMATION TOOL

Identification of the right automation tool is critical to ensure the success of the testing project. The effort put in the tool evaluation process enables successful execution of the project. The selection of the tool depends on the various factors such as:

- Opti freight applications and technology.
- Ease of scripting and reporting capability.
- License cost of tool.
- Detailed comparison matrix.

In ALM tool automation the test scripts for the automation are written. The total test scripts for the automation testing are only 20 while in manual testing these were 68 [5, 6, 7, 8].

Table 2. Application Levels Test Coverage 2

S. No	Application Under test	Number of Automatic Test Scripts
1	Customer Portal	4
2	Vendor Portal	2
3	Lean	3
4	LTL	2
5	Pricing	1
6	Orion	3
7	VRA	2
8	GMS	1
9	WUMQ	2
	TOTAL	20

We have written test scripts and run these test scripts. In ALM tool for automation testing we written some test scripts. The test script creation view is given in figure 4.

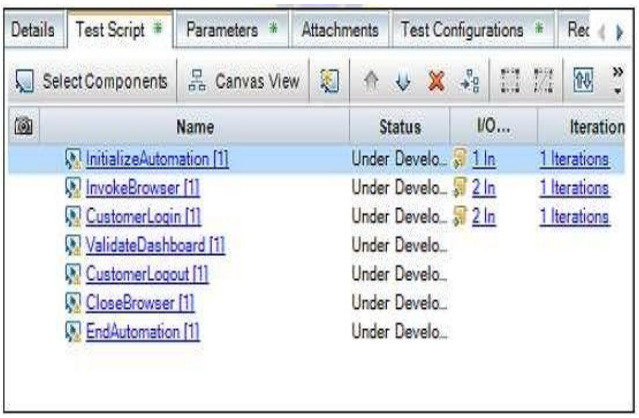

Fig 4. Test Script creation view

The framework architecture for the running test result for the automation testing using ALM tool is given in figure number 5. In figure number 5 all the three main phases are test resources, test plan and test lab. This figure shows all the three phases with test scripts. How these phases are connected to each other.

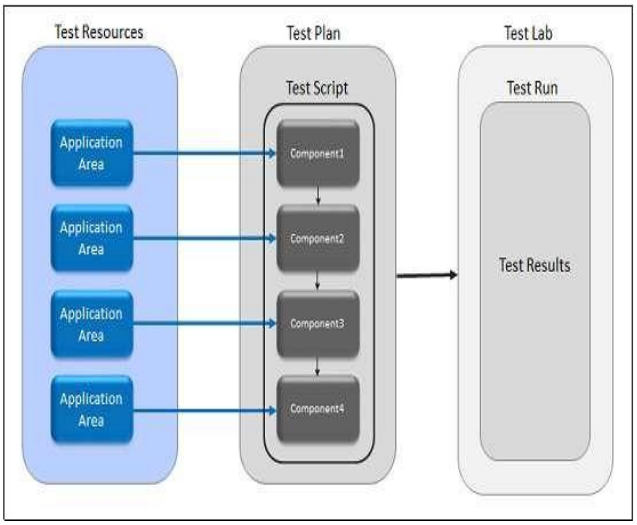

Fig 5. BPT framework architecture

V. AUTOMATIC SCRIPT RUN RESULT

Estimation of automation tools is an important aspect that can be considered when we plan to use an automation tool. For the automation test scripts implementation of tool infrastructure and training of the automation tool is cost effective [8, 9, 10, 11, 12, 13]. So, more scripts for automation is so much more cost effective due to all the aspects that came in during the start of the purchasing of automation tools to delivery of the web application to the customer.

When automation of ALM is run for the scripts then the result for the automation of testing is shown in figure 6. Compared to manual testing the automatic testing is so much suitable for some type of testing. There are so much testing for each function of our web application. These testing are functional testing, structural testing, regression testing, performance testing, load testing, hardware testing etc.

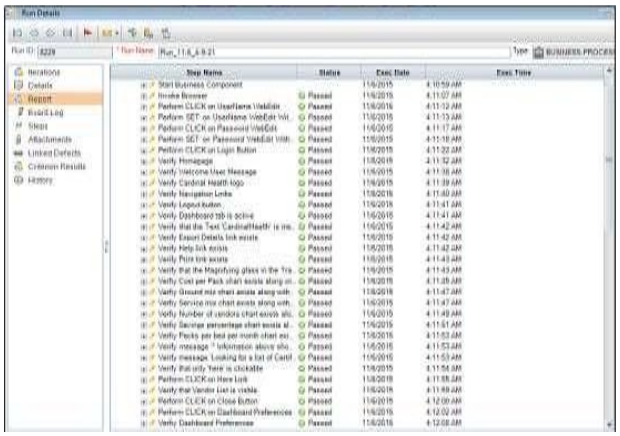

Fig 6. Script run result

In figure 7 we have compared manual testing with automated testing. How the different types of the testing are suitable for our application compare to manual and automated.

Suitability for Testing	Manual	Automated
Helping the development effort with smoke tests and unit tests	●	●●
Functional testing	●●	●●
Structural testing	●	●●
Automation of test input generation	○	●●
Installation and configuration testing	●●	●
Regression testing	●●	●●
Hardware testing (e.g., weigh balance interfaces)	●●	○
Load and performance testing	○	●●
Testing for race conditions	○	●●
Endurance/longevity testing	○	●●
Key: ●● = suitable ● = may be suitable ○ = not suitable		

Fig 7. Suitability of manual and automated testing

VI. CONCLUSIONS

A key decision significantly impacting the test strategy as well as the cost calculation. Which types of test cases will be executed and out of these how much will be automated? Maybe some tests are not suitable for testing and some are more suitable for automation than others. Some test cannot be automated at all, whereas other tests rely on test automation and cannot be executed manually. For all the applications the test scenario starts with manual testing and goes automated with test cases of manual testing. Automation Testing is the current market trend. Time and money are critical to sustain and lead in today's market competition. Automation is faster and more reliable than manual, as it is performed by tools. Though initial investment is needed to procure license for tools, but in the longer run, returns are good for web application [14, 15].

ACKNOWLEDGEMENT

We would like to express our gratitude to Deanship of Scientific Research, King Khalid University, Kingdom of Saudi Arabia for funding this work.

REFERENCES

[1] Zhu, Kunhua, Junhui Fu, and Yancui Li. "Research the performance testing and performance improvement strategy in web application." 2010 2nd International Conference on Education Technology and Computer. Vol. 2. 2010.

[2] Hislop, Helen, Dale Avers, and Marybeth Brown. Daniels and Worthingham's muscle testing: Techniques of manual examination and performance testing. Elsevier Health Sciences, 2013.

[3] Beizer, Boris. Black-box testing: techniques for functional testing of software and systems. John Wiley & Sons, Inc., 1995.

[4] Khan, Rijwan, and Mohd Amjad. "Web application's performance testing using HP LoadRunner and CA Wily introscope tools." 2016 International Conference on Computing, Communication and Automation (ICCCA). IEEE, 2016.

[5] Kit, Edward, and Susannah Finzi. Software testing in the real world: improving the process. ACM Press/Addison-Wesley Publishing Co., 1995.

[6] Hetzel, William C., and Bill Hetzel. The complete guide to software testing. John Wiley & Sons, Inc., 1991.

[7] Kuhn, D. Richard, Dolores R. Wallace, and Albert M. Gallo Jr. "Software fault interactions and implications for software testing." Software Engineering, IEEE Transactions on 30.6 (2004): 418-421.

[8] Khan, Rijwan, and Mohd Amjad. "Automatic generation of test cases for data flow test paths using K-means clustering and generic algorithm." International Journal of Applied Engineering Research 11.1 (2016): 473-478.

[9] Verbauwhede, Ingrid, Patrick Schaumont, and Henry Kuo. "Design and performance testing of a 2.29-GB/s Rijndael processor." Solid-State Circuits, IEEE Journal of 38.3 (2003): 569-572.

[10] Weyuker, Elaine J., and Filippos I. Vokolos. "Experience with performance testing of software systems: issues, an approach, and case study." IEEE transactions on software engineering 12 (2000): 1147-1156.

[11] Ward, C. L., et al. "Design and performance testing of quantitative real time PCR assays for influenza A and B viral load measurement." Journal of Clinical Virology 29.3 (2004): 179-188.

[12] Menascé, Daniel. "Load testing of web sites." Internet Computing, IEEE 6.4 (2002): 70-74.

[13] Zhao, Nai Yan, and Mi Wan Shum. "Technical Solution to Automate Smoke Test Using Rational Functional Tester and Virtualization Technology."Computer Software and Applications Conference, 2006. COMPSAC'06. 30th Annual International. Vol. 2. IEEE, 2006.

[14] DeMillo, Richard A., et al. "An extended overview of the Mothra software testing environment." Software Testing, Verification, and Analysis, 1988., Proceedings of the Second Workshop on. IEEE, 1988.

[15] Mahajan, Manish, Sumit Kumar, and Rabins Porwal. "Applying genetic algorithm to increase the efficiency of a data flow-based test data generation approach." ACM SIGSOFT Software Engineering Notes 37.5 (2012): 1-5.