Reno Test Automation Tool

Shweta[1], Dr. Rajesh Yadav[2]
[1]M. Tech, [2]Associate Professor
Computer Science Engineering (Cse)
Ganga Institute Of Technology And Management, Kablana, Jhajjar , Haryana

Abstract- Software testing is an integral phase in Software Development Life Cycle (SDLC) process. Testing assesses the functionalities of a software item and quality of the product. Automated software testing utilizes different tools to execute testing activities. In this paper, I have discussed the features of automated and manual testing as well as analyzed three automated software testing tools: Selenium, UFT/QTP and WATIR. In brief, I have presented a detailed description focusing on multiple feature set, efficiency, simplicity and usability of each tool. I also evaluated, tested and compared the different aspects of Selenium, UFT/QTP and WATIR. Finally, this research allowed me to draw some solid differences between automated and manual testing as well as learn and explore various characteristics of automated testing tools by having real-world experience of testing effectively. According to the daily use, Mobile and Web applications take the first place in development and testing. Testing automation enables developers and testers to easily automate the entire process of testing in software development saving time and costs. This paper provide a feasibility study for commercial and open source web testing tools helping developers or users to pick the suitable tool based on their requirements.

Keywords- Black Box Testing; web testing tools; open source; Commercial

I. INTRODUCTION

Software end users are more informed and demanding than before. The quality of software determines the success of any software product. This provides a tremendous opportunity for software quality assurance in software industries and that is driven by the customer satisfaction. Developing quality and defect-free products under time and budget constraints have become crucial. To implement such products, with minimum or no error is very difficult, that's why the idea of software testing has come into existence [1]. Software testing has become an essential and extensive activity in the software industry. Testing is the critical part of software development process and indicates the eventual review of the specification, design, and coding [2]. Nowadays, probably no single company exists without performing software testing. Software testing is a method, which is executed for evaluating the functionality and the correctness of the software product, to determine whether it meets the expected features and quality. Testing is an essential phase and an inevitable part of Software Development Life Cycle (SDLC).

II. OBJECTIVE OF RESEARCH

The goal of this research paper is to introduce the various features and presentations of software testing tools (Selenium, WATIR, and UFT/QTP) as well as assess and compare these tools to determine their usefulness. This study also allows drawing the basic comparison among automated and manual testing to represent the significance of automated testing in software industries. To accomplish the goal of this research, the following steps are to be performed.
• Identifying a set of tools to be evaluated.
• Selecting the target application to be tested by all these tools.
• Testing the target application using the selected automated testing tools and gather resulting data.
• Developing a set of comparisons to be used to assess the tools.
• Performing an analysis of each tool and comparing each other based on an ideal feature set and depending on the result of test execution.
• Outlining inferences and making recommendations based on the outcome of research.

III. RELATED WORK

Last researches interested in comparing the capabilities of the testing tools by practicing them or only based features each tool support. Harpreet Kaur, Gagan Gupta conduct a comparative study among selenium, Test Complete and QTP tools the study include many aspects but not drag the automation features of tools such as record and play-back, cross platform or browsers support features [7]. Abha Jain, Manish Jain, Sunil Dhankar[8] compare two commercial tools Ranorex vs. QTP including many features but the main concern on the cost of the total project and the study not include any open source software to compare against. Angmo, R and Sharma, M [9] compare the performance of selenium web driver against watir-web driver the two open source software, Study includes performance parameters such as execution speed which vary in the type of tested Controls. This research is efficient but require more than one tool to give the best judge to the user.

Selenium is a portable framework for testing web applications. Selenium provides a playback (formerly also recording) tool for authoring functional tests without the need to learn a test scripting language (Selenium IDE). It also provides a test domain-specific language (Selenese) to write tests in a number of popular programming languages,

including C#, Groovy, Java, Perl, PHP, Python, Ruby and Scala. The tests can then run against most modern web browsers. Selenium deploys on Windows, Linux, and macOS platforms. It is open-source software, released under the Apache 2.0 license: web developers can download and use it without charge.

IV.   MOBILE TESTING

Mobile Testing is a process by which application software developed for mobile devices is tested for its functionality, usability and consistency. The APP must be tested over different combination of platforms, operating systems and networks before launching to global.

REXA supports testing of mobile applications built for the Android and IOS operating system. REXA offers lots of test operations that can simulate a rich set of user activities and performs other test-related actions:

• Touch and long touch on-screen controls.
• Drag and swipe items.
• Simulate multi-touch gestures.
• Input text via a keyboard.
• Press on-device (physical) buttons.
• Install and run applications on devices.
• Verify objects and their properties.

A.   Mobile Platforms

Let's have a look at following platforms which mainly includes the OS of Mobile, type of device, and type of mobile app. Android, iOS and Windows are the mainly used platform. By using the REXA, Application can be automated on these platforms.

B.   Statement of confidentiality

The recipient of this document agrees to keep this document confidential and not to use its contents in any form or by any means without permission from RenovITe. Moreover, the recipient agrees not to place this document at the disposal of any third party without written permission from RenovITe. If such permission is granted, it will be subject to the condition that the recipient ensures that any third-party recipient of this document, or of information contained therein, is held responsible to RenovITe for the confidentiality of that document or information.

If you believe you are not the intended recipient, please inform RenovITe by email on info@renovite.com.

C.   Mobile Automation Scope

Following types of application can be automated with the help of REXA -

Web Application - Web apps are actually websites that open in your smartphone with the help of a web browser. These are not Real applications. Mobile websites have the broadest audience of all the primary types of applications because of its benefits like Easy access, easy development, easy update and no installation is required as compare to native or hybrid applications.

Native Application –Native applications are developed for one platform which can be installed through Google Play store or Apple's App Store. We can consider Facebook and WhatsApp as an example of native application. Such kind of applications maintain UI design which is fit with each Operating System.

Hybrid Application- Hybrid Applications are combination of Native and Web Applications. Main use of such applications to expose content from existing websites in App Format. Instagram and Wikipedia are the example. Developing the Hybrid app is cheaper than Native app and Maintenance is low.

D.   Mobile Testing Tool

REXA is compatible with following testing tools to test functionality of the mobile applications and to perform regular regression testing.

Emulators - An emulator is a hardware device or a program that pretends to be another particular device or program that other components expect to interact with

Simulators –A mobile simulator is a software application for a personal computer which creates a virtual machine version of a mobile device, such as a mobile phone, iPhone, other smartphone, or calculator, on the computer. This may sometimes also be termed an emulator

Using REXA, Automation can be done on emulators and simulators both. Also, it provides the integration with real devices. REXA has certain configurations before launching the application on these tools which we will be learning in next topics.

E.   How to create Object Repository in REXA Need to Refer 1.x .

How to write Mobile Scripts in REXA

To write a script in REXA, let's understand first what is Script? Script is a process of conversion of manual test cases into automated steps. These are a set of instructions that will be performed on the system to test thefunctionality of the application and make sure it is working as expected.

On Left Navigation in REXA, you can find out the Script option under User Interface Module. Two Tab will be displayed after clicking on Script link– Dashboard and Script.

1-Dashboard Tab is the snapshot of all the created scripts which a user has created so far. It includes following details –

Script name – Name given by user as per test case

Status – Draft, Review (if review happened) , Commit(if it finally accepted)

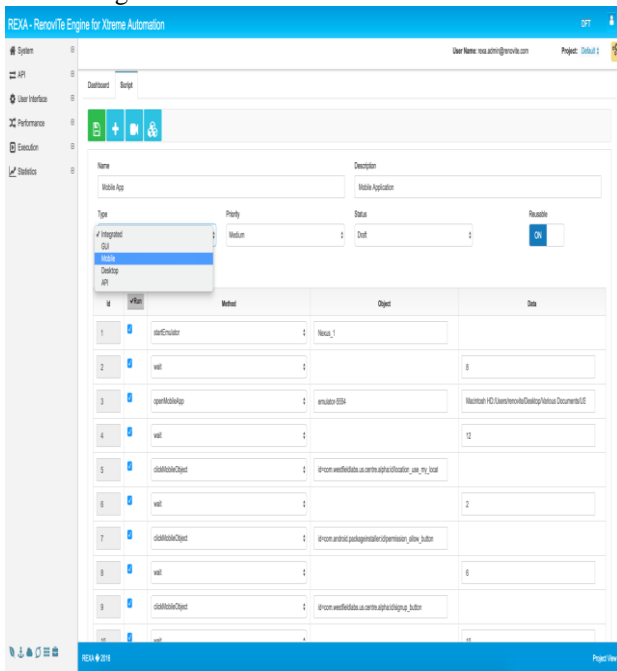Created by – person id who created this script

Created on- Creation Date with Timestamp

Action – Edit(If any updation/modification is required, click on it)

Please see below screenshot for details.

2-Script Tab is the place from where user can write his first script. Let's have a look at following screenshot for better understanding.

Imp: - While writing the Mobile script**,** First Method should be 'StartEmulator' and Object should be containing Device name like Nexus. To Open Mobile application, There is method called as 'openMobileApp' and Object would be i.e. 'emulator- 5554' . Data column would contain the proper path where. apk of the application has been placed. Please see the screenshot for first 3 steps.

a. From above screenshot, Name field is used to define the Script name and Description is used to define the purpose of the script.

b. REXA has provided additional feature by adding the Type Drop down. If user select Mobile option, only Mobile related methods will be displayed in Methods Dropdown list. It has got 5 options– Integrated, GUI, Mobile, API, and Desktop. Please refer Section "Mobile Methods in REXA" to get complete list of methods with description related to Mobile.

c. Priority can be defined by selecting the appropriate value from Priority drop down. It has three values as High, Medium and Low.

d. Status Drop down has three values – Draft, Review and Commit. Draft is the initial status of the script which means you are still developing the script. Once it is done it can be changed to commit as Final one.

e. REXA has provided an important feature as Reusable. if you turn the indicator ON that means this script can be called into another script as it has been marked as reusable now. By Default, it is OFF.

f. Id is the Sequence column.

g. Run Column is checked by Default and tells that this steps will be part of execution. User can uncheck it if any steps they don't want to be part of execution.

h. Next Column is Method. Methods are actions you want to perform to achieve a particular behaviour. For example, if you want to open a url then you can select the method as "openUrl". Please refer the List of all methods given in appendix "Methods".

i. Next Column is Object. Objects are parameter which would be used in relation to the method selected. For example, if the method is "open Browser" that means we want the browser to be open when browser is launched, another example would be if user uses method as type so parameter would be locator of the UI element where user wants to enter the text.

j. Next Column is Data where user can enter the data pertaining to Method/Objects selected in the same row.For example, if the method is "open Browser" so that parameter (data) it would take is what browser you want to launch Firefox/IE/Chrome, another example would be if user uses method as type so parameter would be what data you would want to type in the located element.

k. Once Script is ready, Click to save the script.

l. To add extra steps press the plus icon.

TIPS –
1-Record Option within Script- REXA provides the recording

functionality also from this page itself. Click on to start the recording mode and start recording the actions performed on an application.

2- User can select Object Repository option [icon] also from Script page itself.

List of locators which can be used are defined in appendix "Locators". REXA has additional feature for the object field using which user can use pre-defined functions, environment variables, user defined test data, database connection. Examples for usage of various types are defined in appendix "Objects/Data"

Edit/Delete a Script- Once script is ready and user wants to make any changes to script He/she can edit the script from dashboard where he/she needs to identify the desired script and click on pencil [icon]

Icon. it will redirect to Script Tab in order to allow him to make any changes. To Delete any Script from Dashboard, Right Click on script and select Delete Option.

F. Technology Stack
- Framework

    - Spring

    - Hibernate

- Database

    - MySQL

- Configuration

    - .Properties

    - .xml

- JPOS – ISO Requests

- LOG4J

G. High Level Architecture
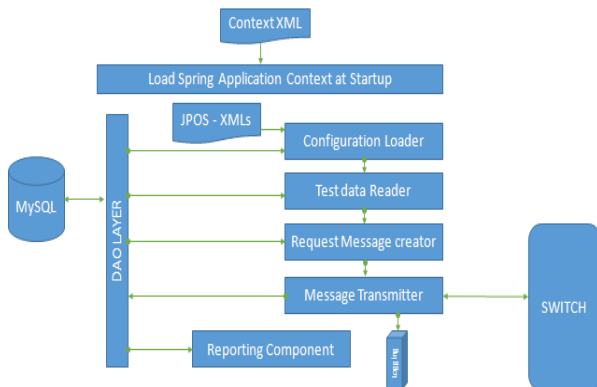
H. Modules
- **Configuration Loader**: Loads various configuration such as switch URL, port, Channel, MUX settings into application context. Interacts with DB to load configuration.
- **Test Data Reader:** Reads the test data such as type of transaction, fields and their values.

- **Request Message Creator**: Populates the fields of the message with their respective values received from Test Data Creator Module.

- There will be 2 types of Message creator implementations :
    - **ISO Message Creator** → For creating ISO messages.
    - **Http Message Creator** → For creating http messages.

- **Message Transmitter**:
    - Receives the formatted message from Message Creator Module and forwards it to switch based on the protocol requested. Subsequently passes the request to DAO to be saved in DB and to the Logging module.

    - Receives the response and passes it on to DAO and Logger.

- **Execution Report Creator**: Reads the Database and creates a formatted execution report.

I. Configurations
- SWITCH Connection parameters
- Type of Transactions
- Values of request message fields.
- Request mode – ISO/Http
- Request type – Asynchronous/Synchronous.
- Encrypted/Unencrypted Request.
- Request templates : ISO/HTTP
- ISO Spec
- Parameters of DB where logging/configuration will be stored.

J. Streamsets
StreamSets is an open source, enterprise-grade, continuous big data ingest infrastructure that accelerates time to analysis by bringing unprecedented transparency and processing to data in motion.
StreamSetsData Collector is a lightweight, powerful engine that streams data in real time. Use Data Collector to route and process data in your data streams.
To define the flow of data for Data Collector, you configure a *pipeline*. A pipeline consists of stages that represent the origin

and destination of the pipeline, and any additional processing that you want to perform. After you configure the pipeline, you click Start and Data Collector goes to work.

Data Collector processes data when it arrives at the origin and waits quietly when not needed. You can view real-time statistics about your data, inspect data as it passes through the pipeline, or take a close look at a snapshot of data.

K. Data Collector Console

Data Collector provides a console to configure pipelines, preview data, monitor pipelines, and review snapshots of data. The Data Collector console includes the following general areas and icons:
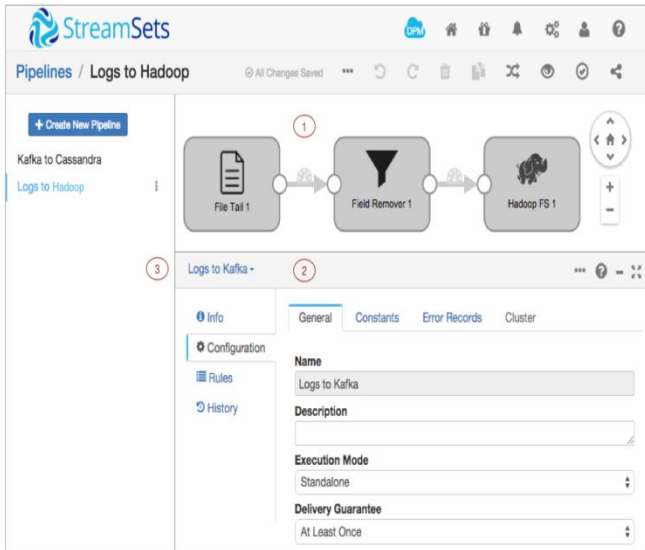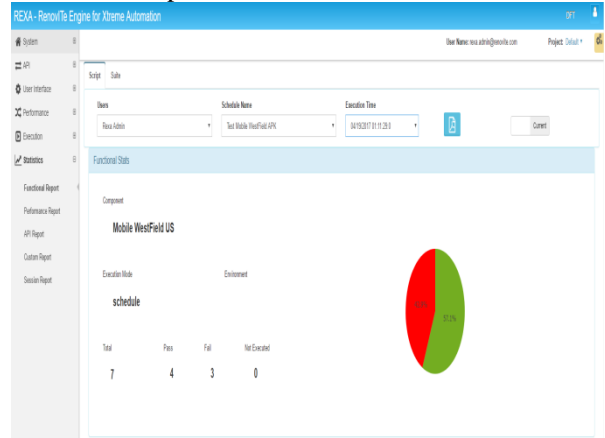

L. Statistics

Execution reports provide information about the status of the test effort, including overall software quality and test execution progress against the planned progress and Reporting is most important part of any tool. Once execution is over and user wants to view the report on how many scripts are passed or failed. For this purpose, REXA has provided the statistics section which is divided into fivemain categories –Execution Report,Host Report and Session Report.

M. Functional Report-

It will show three tabs once clicked from Left Navigation. These tabs are Data, Script and Suite. User can easily navigate to any tab and can view the execution report. If we talk about Script Tab, it will show Users drop down, Schedule Name and one indicator. Select the user name from User Drop down and appropriate schedule name from Schedule drop down. Next to these option, you will find one another indicator and by default it is Current selected which means current instance will have displayed in terms of report but you can change this

indicator to 'ALL' which means all report of that instance of script/Data/Suite will be displayed.

Graphical Representation- Let's have a look at below screenshot which will give you clear idea on Execution Reporting module as how it looks like. Graphical Representation is displayed here along with Pass, Fail and Total count of scripts.

Performance Statistics–(TBD)Refer the second screenshot in continuation to above one. It is showing the performance statistics data which has been built by considering the Total transactions, total time taken and calculating the average transaction time. It also shows the step wise details and user can easily come to know which step is failed and what was the reason for failure.
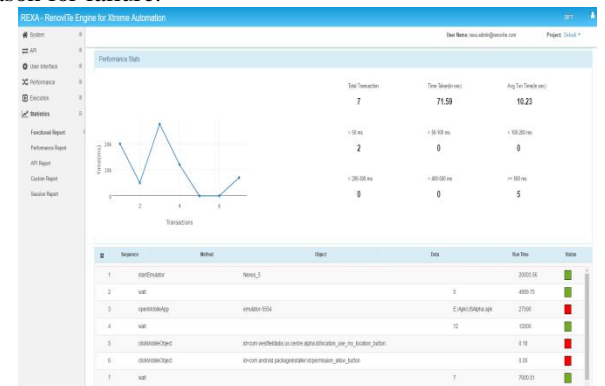
REXA is going to provide more flexibility to User to customize the report based on requirement and detailed report with multiple options.

N. Case Study - Exercise

Now It's time to do some exercise on whatever we have covered in this training tutorial document by doing following exercise –

Case 1 –Let's create one Mobile Script which will run on Emulator and make sure following REXA modules will be

part of integration – Suite, Data, Scheduler. …………………………………………. Etc .

V.   CONCLUSION AND FUTURE WORK

Our research work comprises of the analyzation of different automated web testing tools for not also commercial but also involve open source tools. This study helping in selecting the suitable tools based on multiple criteria. Selecting tools in this area, it is important to consider multiple parameters which vary among different requirements, many requests in the market make the cost the first target to be considered, in the other hand some open sources software didn't provide support for its user as it work under user experience .The study present each tools with features which in the same and different degree with other tools and how each tool behave against others tools' features .This comparative study can be the basis for developing a model facilitate selecting the most applicable tools based on the needed requirements.

Our future work will encounter more tools and more features also that will help in building a user based requirement model. This model also will help researches to select tools helping their research work.

VI.   REFERENCES

[1]. Ms. Shikha maheshwari1 „A Comparative Analysis of Different types of Models in Software Development Life Cycle" International Journal of Advanced Research in Computer Science and Software Engineering Volume 2, Issue 5, May 2012.

[2]. Boydens, Jeroen. Location transparency and transactions as first-class concepts in object-oriented programming languages. Diss. PhD thesis, KU Leuven, 2008.

[3]. Bellettini, Carlo, Alessandro Marchetto, and Andrea Trentini. "TestUml: user-metrics driven web applications testing." Proceedings of the 2005 ACM symposium on applied computing. ACM, 2005.

[4]. Prof. (Dr.) V. N. Maurya, Er. Rajender Kumar "Analytical Study on Manual vs. Automated Testing Using with Simplistic Cost Model",International Journal of Electronics and Electrical Engineering ISSN:2277-7040 Volume 2 Issue 1 (January 2012).

[5]. Jomeiri, Alireza. "A SURVEY ON WINDOWS-BASED WEB TESTING TOOLS." International Journal of Academic Research 6.4 (2014).

[6]. Binder, Robert. Testing object-oriented systems: models, patterns, and tools. Addison-Wesley Professional, 2000.

[7]. Harpreet kaur et al Int. Journal of Engineering Research and Applications ISSN : 2248-9622, Vol. Issue 5, Sep-Oct 2013, pp.1739-1743

[8]. Abha Jain, Manish Jain, Sunil Dhankar International Journal of Engineering, Management & Sciences (IJEMS)ISSN-2348 – 3733, Volume-1, Issue-1, January 2014

[9]. Angmo, Rigzin, and Monika Sharma. "Performance evaluation of web based automation testing tools." Confluence The Next Generation Information Technology Summit (Confluence), 2014 5th International Conference-. IEEE, 2014.

[10]. Bruns, Andreas, Andreas Kornstadt, and Dennis Wichmann. "Web application tests with selenium." Software, IEEE 26.5 (2009): 88-91

[11]. M. Sharma, R. Angmo,"Web based Automation testing and Tools," International Journal of Computer Science and Information Technologies, vol. 5, no. 1, pp. 08-912, 2014.

[12]. Harpreet kaur et al Int. Journal of Engineering Research and Applications ISSN : 2248-9622, Vol. Issue 5, Sep-Oct 2013, pp.1739-1743.

[13]. M. Sharma, R. Angmo,"Web based Automation testing and Tools," International Journal of Computer Science and Information Technononologies (IJCSIT), vol. 5, no. 1, pp. 08-912, 2014.

[14]. Dustin, Elfriede, Jeff Rashka, and John Paul. Automated software testing: introduction, management, and performance. Addison-Wesley Professional, 1999

[15]. Nguyen, Hung Q. Testing applications on the Web: Test planning for Internet-based systems. John Wiley & Sons, 2001.

[16]. Jain, Abha, Manish Jain, and Sunil Dhankar. "A Comparison of RANOREX and QTP Automated Testing Tools and their impact on Software Testing." IJEMS 1.1 (2014): 8-12.

[17]. Dubey, Neha, and Mrs Savita Shiwani. "Studying and Comparing Automated Testing Tools; Ranorex and TestComplete." IJECS 3.5 (2014): 5916-23.

[18]. Nagarani, P., and R. Venkata Ramana Chary. "A tool based approach for automation of GUI applications." Computing Communication & Networking Technologies (ICCCNT), 2012 Third International Conference on. IEEE, 2012.

[19]. Mohd. Ehmer Khan, "Different Forms of Software Testing Techniques for Finding Errors,"IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 3,No 1, May 2010.