

RUNNING CONCEPT OF AUTOMATION TESTING TECHNIQUES

Sweta Dwivedi

M.Tech, Dept. of CSE, NIET College of Engineering, Greater Noida

Abstract - Testing is a general activity in Software Development Process. It is to examine and change source code. Convincing Testing produces choice programming. This Paper deals with a basic and key issue of Software Testing. Testing can be driven physically and furthermore Automated. These Techniques have their own purposes of intrigue and hindrances. The Objective of this paper is to perform Automation Testing using Software Testing Tool "Selenium". With this web testing mechanical assembly, test cases are normally recorded in establishment while analyzer is entering the data in a web application screen.

Key Words: Software Testing, Manual Testing, Automation Testing, Selenium.

1. INTRODUCTION

A definitive objective of programming improvement is to create brilliant programming. Predominant quality programming has attributes like minimal effort, dependable and client palatable. Testing is the way toward executing a program with the goal of discovering blunders. This is a significant and basic movement to find every one of the blunders early programming advancement prepare. Vehement and productive testing lessens the framework cost. Programming improvement houses enroll testing and quality confirmation work force for performing the testing action.

Test means program's reaction to ever conceivable information. A program ought to test for each substantial and invalid info. Testing movement can be led into two ways: Manual testing and computerization testing. Any programming testing sort can be executed both physically also utilizing a computerization apparatus.

Manual testing is in which the testing movement is performed by testing people. Manual testing requires an analyzer to perform manual test operations on the test programming without the assistance of Test computerization. Manual Testing is a procedure where in an analyzer frequently takes after a composed test arrange for that leads them through an arrangement of imperative experiments. An experiment in programming testing is an arrangement of conditions composed for specific applications and analyzer run every one of these conditions to guarantee the right usefulness of the product applications. Manual testing is a relentless action that requires the analyzer to have a specific arrangement of characteristics; to be understanding, attentive, theoretical, inventive, creative, receptive, ingenious, stubborn, and skilful. Keeping in mind

the end goal to completely test that every one of the necessities of an application are met, there must be no less than two experiments for every prerequisite: one positive test and one negative test. Manual testing finds deserts identified with the convenience testing and GUI testing region. Any new application must be physically tried before its testing can be computerized. Manual testing requires more exertion, however, is important to check computerization plausibility. Manual testing does not require the information of any testing instrument.

Mechanization Testing is running experiments where manual mediation is not required to run every one. It utilizes unique programming to compose and execute test cases to contrast the genuine result and the anticipated result. When tests have been computerized, they can be run rapidly and more than once. Mechanized programming testing is an ideal approach to build the adequacy, effectiveness and scope of programming testing. Robotization testing requires the impressive measure of speculation for purchasing the product and good equipment assets. Mechanization testing does what manual testing does not. Computerization testing enhances the precision, and it spares the season of the analyzer and association's cash. Robotization testing is most appropriate in the earth where the prerequisites are as often as possible changing, and a colossal measure of relapse testing is required to be performed. Mechanization testing is most appropriate in nature where there are the experiments that are to be executed over and over. It builds the nature of testing structure and lessens the future upkeep cost. Different advantages of Automation testing are quick keep running of experiment. Reusable experiments are made, and these experiments are dependable, extensive and Programmable.

The primary contrast between Manual testing and Automated testing is that Automation testing is most appropriate for nature where the Repetitive work is more (e.g., running relapse tests, re-entering a similar test information, and checking against coding measures). Likewise, manual testing is most appropriate for nature where the prerequisite changes ceaselessly.

The order of reference in the running text should match with the list of references at the end of the paper.

1 Prologue to Selenium

Selenium is a versatile programming testing utilized for Automation testing It is a structure involves many instruments utilized for testing web applications. Selenium gives a record/playback device for creating tests without

taking in a test scripting dialect (Selenium IDE). It has a test area particular dialect (Selenese) to compose test cases in various mainstream programming dialects, including C#, Java, Groovy, Perl, PHP, Python and Ruby. All these experiments composed can then keep running against most present day web programs. Selenium can keep running on any working framework stages like Windows, Linux, and Macintosh and so forth. It is open source programming that can be down stacked from the Site of selenium. Selenium was initially created by Jason Huggins, who was later joined by different software engineers and analyzers at thought works.

Selenium is a suite of devices for web mechanization testing. Selenium Suite incorporates taking after segments: Selenium IDE, Selenium Core, Selenium 1 (known as Selenium RC or Remote Control), Selenium 2 (known as Selenium Web Driver), Selenium-Grid

2. Selenium IDE

Selenium IDE is a coordinated advancement environment for Selenium scripts. It is executed as a Firefox augmentation, and permits you to record, alter, and investigate tests. It was beforehand known as Selenium Recorder. Selenium IDE is not just recording device: it is a total IDE. Selenium IDE (Integrated Development Environment) works like business devices like QTP, Silk Test and Test Partner and so on.

In this paper our concentration is to perform computerization testing of a web application utilising robotisation testing apparatus "Selenium IDE".

STEPS FOR IMPLEMENTATION OF TEST CASES USING SELENIUM

Problem analysis:

It is the way toward social affair data to find the premise of positives and negatives of a proposed framework. For instance, let us consider an issue to build up a site for client enrollment and login. For this, we have composed a site with the name DEMO. Here we need to test the page of an enrollment type of a site DEMO.

The preview for Demo is:



Fig -1: Main screen of website DEMO



Fig -2: Screen of registration page

Composing experiments: Test case is a report that incorporates a technique to perform testing. An experiment incorporates an arrangement of test data sources, execution conditions, and expected yield produced for a specific goal, e.g. to check a specific program way or to confirm that the particular information will meet with the wanted yield. There is no recommended design for composing an experiment. , however, an experiment must incorporate information, expected conduct, expected yield. We have a specimen experiment for first name field of enrollment page of DEMO site.

TEST CASE OF "FIRST NAME" FOR REGISTRATION PAGE:

Table -1: Test cases written for First Name field of Registration page

Test Case Id	Test Description	Execution Steps	Expected Behavior
First Name	Checking the functionality of the "Name" text box.	1) Entering the value as "12345."and clicking on the "Submit."Push button. 2)Entering the value as "/*-+." and clicking on the "Submit." Push button. 3)Entering the value as "12354/*-+."and clicking on the "Submit." Push button. 4)Entering the value as Null Characters and clicking on the "Submit."Push button. 5)Entering the value as " " blank space and clicking on the "Submit"	1) Should show the popup error message "Invalid entry. First Name accepts only characters".After clicking the ok button in the message the cursor should focus on the textbox. 2)Should show the popup error message "Invalid entry. First Name accepts only characters". After clicking the ok button in the message cursor should focus on the textbox. 3)Should show the popup error message "Invalid entry. First Name accepts only characters".After clicking the ok button in the message the cursor should focus on the textbox. 4)Should show the popup error message "FirstName is a Required field". After clicking the

	<p>Push button</p> <p>6) Entering the value as "a to z" and clicking on the "submit" Push button</p> <p>7) Entering the value as "abcd12354/*-+-" and clicking on the "Submit" Push button</p> <p>8) Entering Infinite entry and clicking on the "Submit" Push button</p>	<p>ok button in the message the cursor should focus on the textbox.</p> <p>5) Should show The popup error Message "Invalid entry. First Name accepts only characters". After clicking the Ok button in the message the cursor should focus on the textbox.</p> <p>6) Should accept the value and cursor Should focus on the "Last Name" textbox.</p> <p>7) Should show The popup error Message "Invalid entry. First Name accepts only characters". After clicking the Ok button in the message the cursor should focus on the textbox.</p>
--	---	---

In the event that mouse pointer is put on the catch then there is a popup message "NOW RECORDING, CLICK TO STOP RECORDING". This message demonstrates that Selenium IDE is prepared for recording experiments.

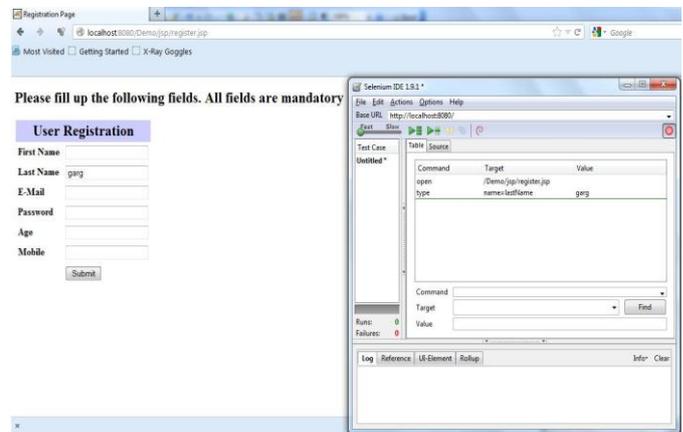


Fig -4 : Screen for entering the value of field Last Name & corresponding record in selenium.

Implementation of test cases using SELENIUM tool

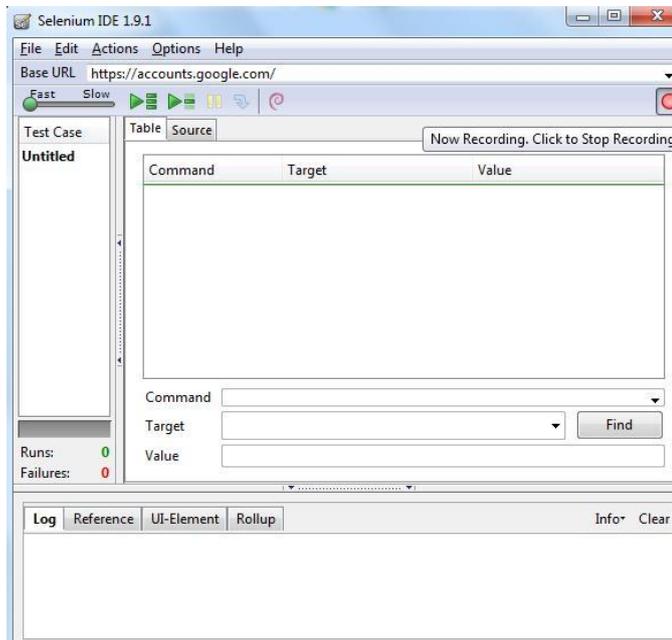


Fig -3: Screen of Selenium for recording Test Case.

Selenium IDE is an extra component in Firefox. To actuate selenium, the accompanying strides are to be taken after. Select apparatuses choice from Firefox menu bar. Select selenium IDE choice from devices popup. The screen appeared in the figure will show up. It has a recording catch on right half of the screen.

(ii) There is a red shading recording catch on the right half of the selenium environment. When initiating recording mode, then what so ever information is entered in the page will consequently recorded in the selenium.

For instance when estimation of Last Name is entered in the site page, the summon relating to that is recorded on the selenium IDE with order "sort" target "name =Last Name" and esteem "garg". Same esteem entered on site page.

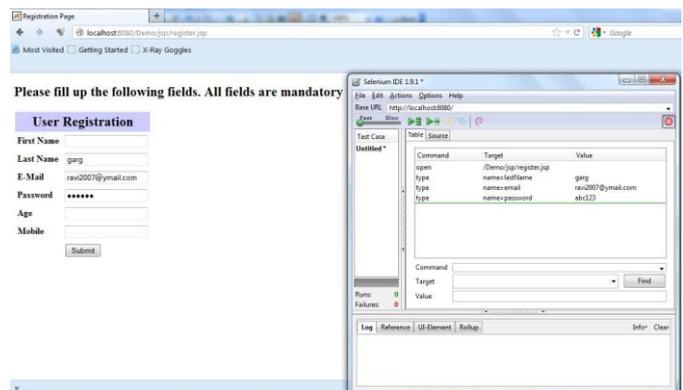


Fig -5: Screen for entering the value of field E-Mail & corresponding record in selenium.

(iii) At the point when estimation, of E-mail is entered in the page, the "ravi2007@gmail.com". Type command corresponds order comparing to that is recorded on the selenium action taken by user, means typing the data. IDE with order "sort" target "name =email" and esteem

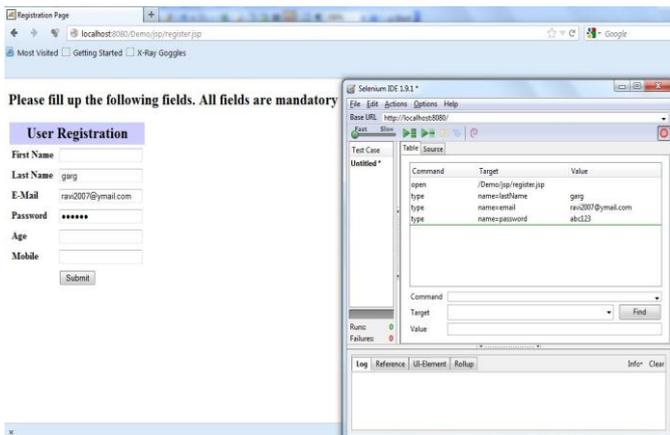


Fig -6: Screen for entering the value of field Password & corresponding record in selenium.

The estimation of the secret key is entered on the website page; the summon relating to that is recorded on the selenium IDE with summon "sort" target "name=password" and esteem "abc123". Sort summon relates move made by client, implies writing the information. "Focus" on selenium screen relates to handling whose esteem is to be executed amid execution of the experiment

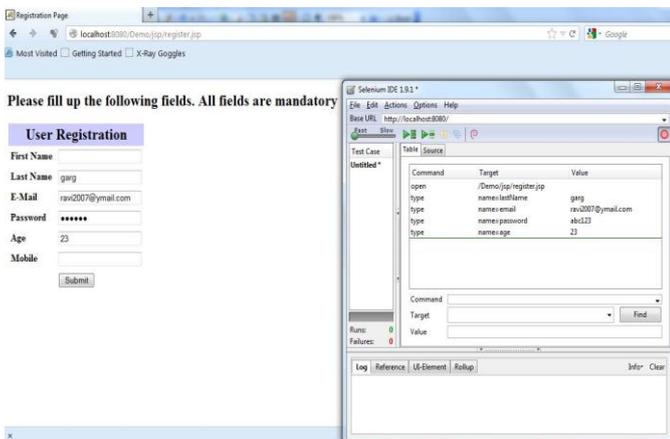


Fig -7: Screen for Screen for entering the value of field Age & corresponding record in selenium.

The estimation of "age" is entered on the website page, the charge relating to that is recorded on the selenium IDE with order "sort" target "name=age" and esteem "23". Sort order compares move made a client implies writing the information."Focus" on selenium screen relates to handling whose esteem is to be executed amid execution of the experiment. "esteem" is the normal yield which ought to coordinate with the genuine esteem amid the keep running the experiment.

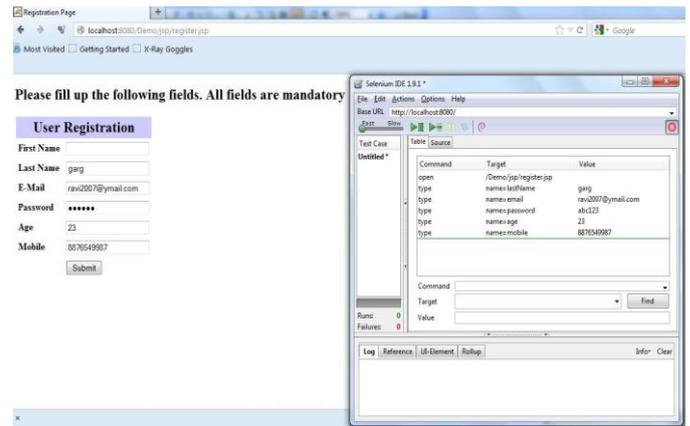


Fig -8: Screen for entering the value of field Age & corresponding record in selenium.

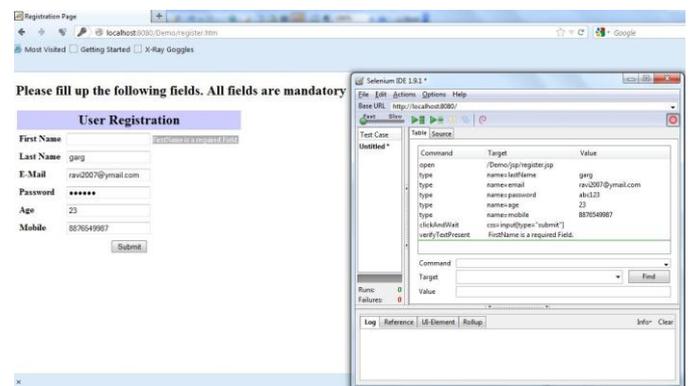


Fig -9: Screen for entering the command "verify text present" in selenium.

After to entering the information, a screen will show up as appeared in figure 9. At that point client needs to quit recording by tapping on red catch and to enter a summon "confirm content present" and target esteem "First Name is a required field". This charge is utilised to look the message a client has not entered the information for the field "first name".

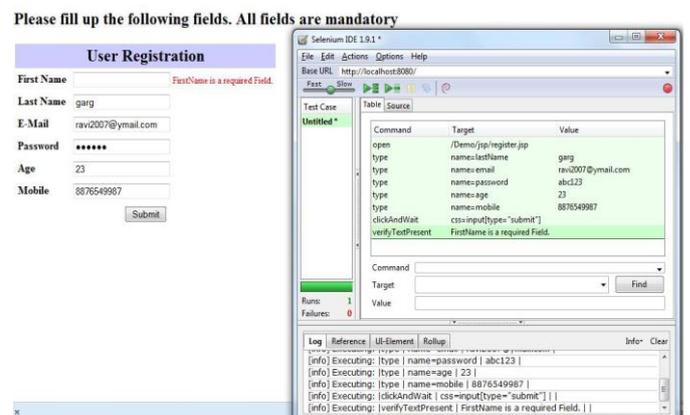


Fig -10: Screen for successful running of test case no. 4 given in Table 1.

Utilizing selenium, when we run the experiment utilizing run catch, then every one of the estimations of the fields are coordinated with the objective values, whose qualities are entered through TYPE summon. Be that as it may, the estimations of the message will be coordinated through Verify content present.

3. CONCLUSIONS

Automated Software Testing is an ideal approach to expanding the viability, effectiveness and scope of programming testing and Selenium is a structure includes many instruments utilized for testing web applications. With the assistance of the contextual investigation, we dissect and discover the testing of a web application utilizing robotization testing apparatus "Selenium IDE". Utilizing this approach, test cases are consequently recorded in foundation while analyzer is entering the information in a web application screen, and these experiments are reusable and most appropriate in the Regression Testing environment.

ACKNOWLEDGEMENT

I thank Prof Pradeep Kumar, Prof. and Head, Dept. of M.tech, NIET College Of Engineering, Greater Noida For his continuous support and encouragement for completing this research paper and also thanks to NIET management.

REFERENCES

- [1] <http://seleniumhq.org/>
- [2] <http://code.google.com/p/selenium/>
- [3] <http://www.guru99.com/automation-testing.html>
- [4] <http://softwarequalitymethods.com/papers/autoarch.pdf>
- [5] http://en.wikipedia.org/wiki/Manual_testing
- [6] http://www.origsoft.com/whitepapers/software-testing-glossary/glossary_of_terms.pdf