STUDY AND ANALYSIS OF AUTOMATION TESTING TECHNIQUES

Vishawjyoti[*] and Sachin Sharma

Deptt of Computer Applications, Manav Rachna International University,

Faridabad

sachin.fbc@mriu.edu.in

Abstract: Testing is a very important activity in Software Development Process. It is to examine & modify source code. Effective Testing produces high quality software. This Paper deals with a significant and vital issue of Software Testing. Testing can be conducted manually as well as Automated. These Techniques have their own advantages & disadvantages. The Objective of this paper is to perform Automation Testing using Software Testing Tool "Selenium". With this web testing tool, test cases are automatically recorded in background while tester is entering the data in a web application screen.

Keywords: Software Testing, Manual Testing, Automation testing, Selenium.

INTRODUCTION

The ultimate goal of software development is to produce high quality software. Superior quality software has characteristics like low cost, reliable and user satisfactory. Testing is the process of executing a program with the intention of finding errors. This is a crucial & essential activity to discover all the errors early software development process. Emphatic & fruitful testing reduces the system cost. Software development houses recruit testing & quality assurance personnel for performing testing activity.

Test means program's response to ever possible input. A program should test for every valid & invalid input. Testing activity can be conducted in to two ways: Manual testing & automation testing. Any type of software testing type can be executed both manually as well using an automation tool.

Manual testing[5] is in which the testing activity is performed by testing persons. Manual testing requires a tester to perform manual test operations on the test software without the help of Test automation. **Manual Testing** is a process where in a tester often follows a written test plan that leads them through a set of important test cases. A test case in software testing is a set of conditions written for particular applications & tester run all these conditions to ensure the correct functionality of the software applications. Manual testing is a laborious activity that requires the tester to possess a certain set of qualities; to be patient, observant, speculative, creative, innovative, open-minded, resourceful, opinionated, and skilful. In order to fully test that all the requirements of an application are met, there must be at least two test cases for each requirement: one positive test and one negative test. **Manual testing** helps discover defects related to the usability testing and GUI testing area. Any new application must be manually tested before its testing can be automated. Manual testing requires more effort, but is necessary to check automation feasibility. Manual testing does not require the knowledge of any testing tool.

Automation Testing [6] is running test cases where manual intervention is not required to run each one. It uses special software to write & execute test cases to compare the actual outcome with the predicted outcome. Once tests have been automated, they can be run quickly and repeatedly. **Automated software testing** is the best way to increase the effectiveness, efficiency and coverage of software testing. Automation testing requires considerable amount of investment for buying the software & compatible hardware resources. Automation testing does what manual testing does not. Automation testing improves the accuracy & it saves the time of the tester & organization's money. Automation testing is best suited in the environment where the requirements are frequently changing & huge amount of regression testing is required to be performed. Automation testing is best suited in the environment where there are critical test cases that are to be executed repeatedly. It increases the quality of testing structure & reduces the future maintenance cost. Various benefits of Automation testing are fast run of test case. Reusable test cases are made & these test cases are reliable, comprehensive & Programmable.

The main difference between Manual testing & Automated testing is that Automation testing is best suited for the environment where the Repetitive work is more (e.g., running regression tests, re-entering the same test data, and checking against coding standards). Also, manual testing is best suited for the environment where the requirement changes continuously.

INTRODUCTION TO SELENIUM

Selenium[1] is a portable software testing used for Automation testing It is a framework comprises of many tools used for testing web applications. Selenium provides a record/playback tool for authoring tests without learning a test scripting language (**Selenium IDE**). It has a test domain-specific language (**Selenese**) to write test cases in a number of popular programming languages, including C#, Java, Groovy, Perl, PHP, Python and Ruby. All these test cases written can then be run against most modern web browsers. Selenium can run on any operating system platforms like Windows, Linux, and Macintosh etc. It is open source software that can be down loaded from the

website of selenium. Selenium was originally developed by Jason Huggins, who was later joined by other programmers and testers at thought works.

Selenium is a suite of tools for web automation testing. Selenium Suite includes following components: Selenium IDE, Selenium Core, Selenium 1 (known as Selenium RC or Remote Control), Selenium 2 (known as Selenium Web driver), Selenium-Grid

SELENIUM IDE

Selenium IDE is an integrated development environment for Selenium scripts. It is implemented as a Firefox extension, and allows you to record, edit, and debug tests. It was previously known as Selenium Recorder. Selenium IDE is not only recording tool: **it is a complete IDE**. Selenium IDE (Integrated Development Environment) works similar to commercial tools like QTP, Silk Test and Test Partner etc.

In this paper our focus is to perform automation testing of a web application using automation testing tool "Selenium IDE".

STEPS FOR IMPLEMENTATION OF TEST CASES USING SELENIUM

Problem analysis:

It is the process of gathering information to discover the basis of positives & negatives of a proposed system. For example let us consider a problem to develop a website for user registration & login. For this we have designed a website with the name DEMO. Here we have to test the web page of a registration form of a website DEMO. The snapshot for Demo is
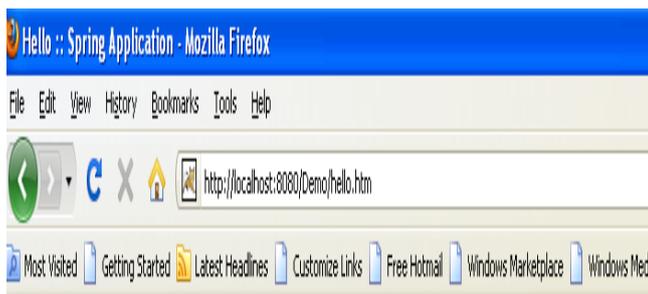
Figure 1: Main screen of website DEMO
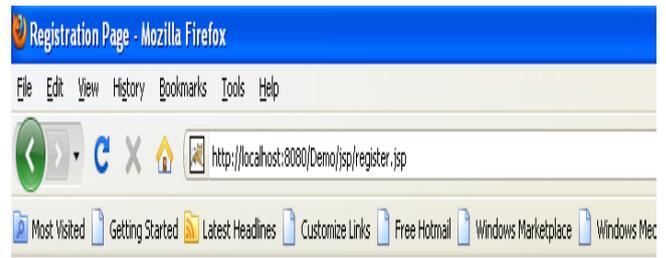
Figure 2: Screen of registration page

Writing test cases: Test case is a document that includes a procedure to perform testing. A test case includes a set of test inputs, execution conditions, and expected output developed for a particular objective, e.g. to check a particular program path or to verify that the specific input will meet with the desired output. There is no prescribed format for writing a test case. , but a test case must include input, expected behavior, expected output. We have a sample test case for first name field of registration page of DEMO website.

TEST CASE OF "FIRST NAME" FOR REGISTRATION PAGE:

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

S.No	Test Case Id	Object	Test Description	Execution Steps	Expected Behavior
01	First Name	Text box	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" Push button 6. Entering the value as "a to z" and clicking on the "submit" Push button 7. Entering the value as "abcd12354/*--+-" and clicking on the "Submit" Push button 8. Entering Infinite entry and clicking on the "Submit" Push button	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 the 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 ok button in the message the cursor should focus on the textbox. 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. 6. Should accept the value and cursor should focus on the "Last Name" textbox. 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. 8. Should able to

					accept up to "35" Characters

Implementation of test cases using SELENIUM tool
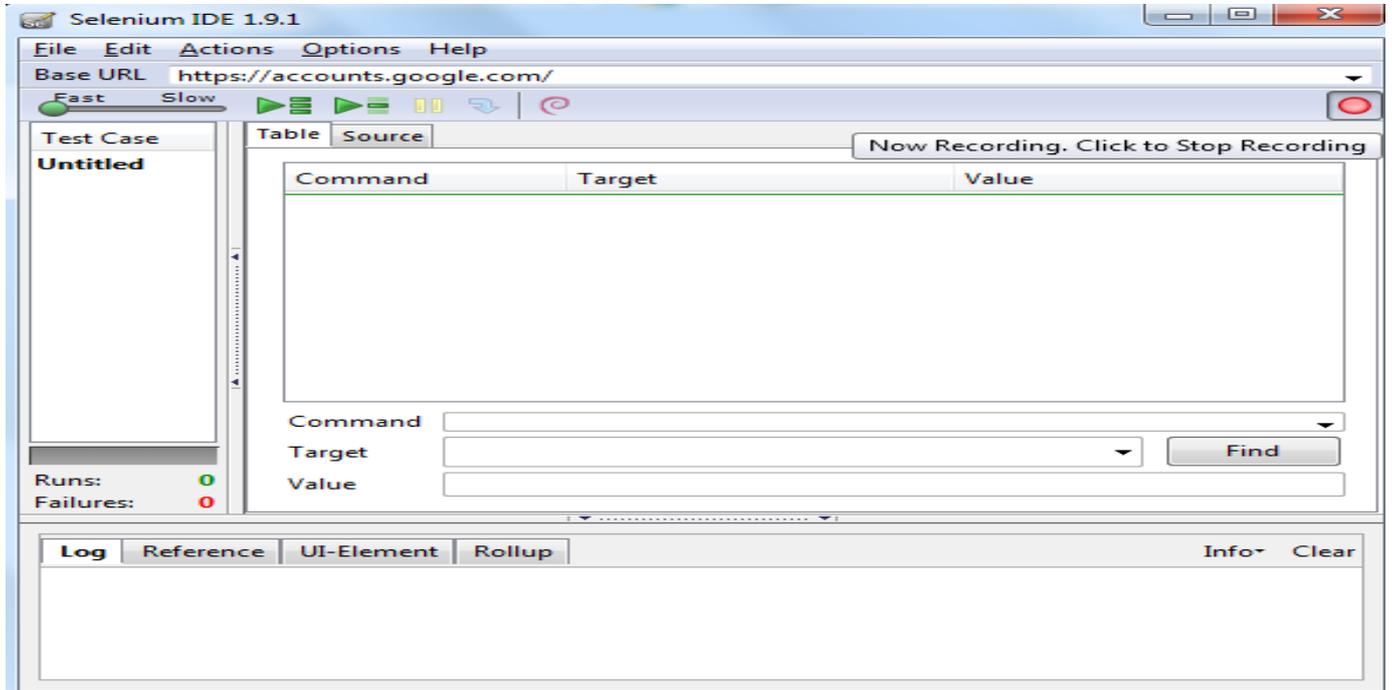
Figure 3: Screen of Selenium for recording Test Case.

Selenium IDE is add on feature in Firefox. To activate selenium the following steps are to be followed.

Select tools option from firefox menu bar. Select selenium Ide option from tools popup. The screen shown in figure will appear. It has a recording button on right side of the screen. If mouse pointer is placed on the button then there is a popup message " NOW RECORDING ,CLICK TO STOP RECORDING". This message shows that selenium ide is ready for recording test cases.
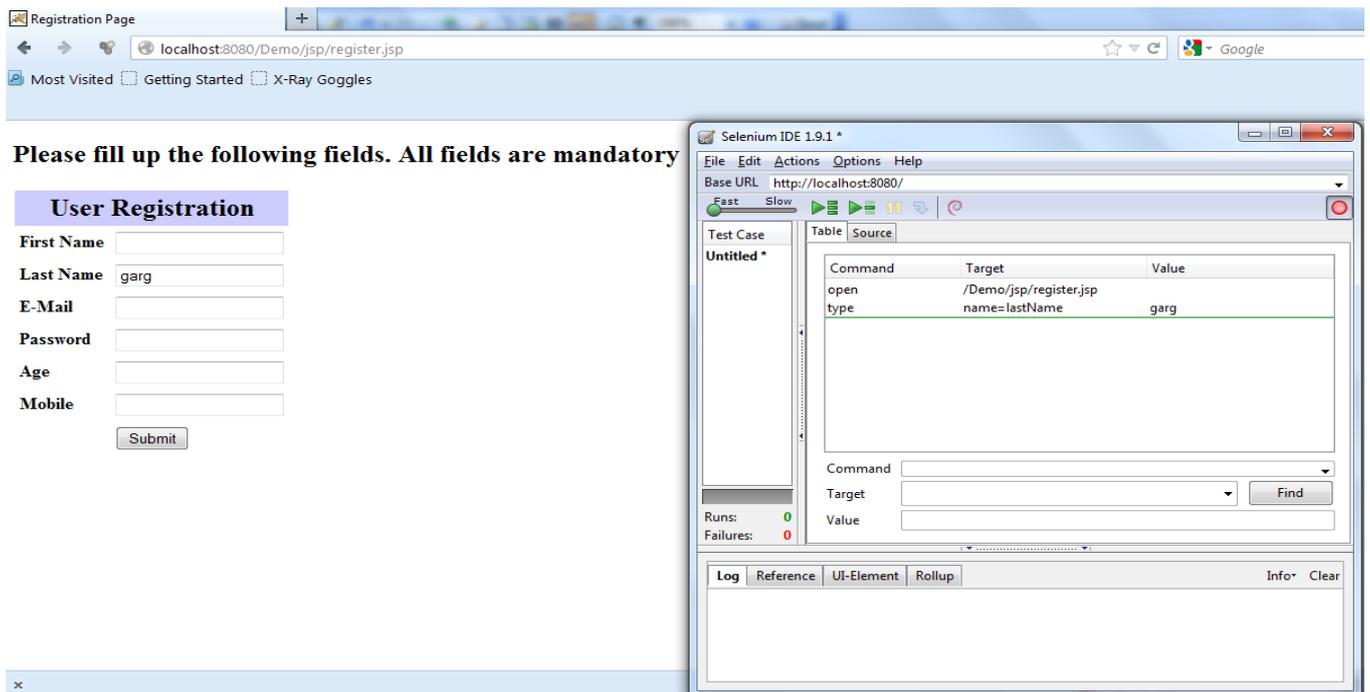
Figure 4: Screen for entering the value of field Last Name & corresponding record in selenium.

(ii) There is a red color recording button on the right side of the selenium environment. Once activating recording mode, then what so ever data is entered in the web page will automatically recorded in the selenium. For example when value of Last Name is entered in the web page, the command corresponding to that is recorded on the selenium IDE with command "type" target "name =LastName" and value "garg". Same value entered on web page.
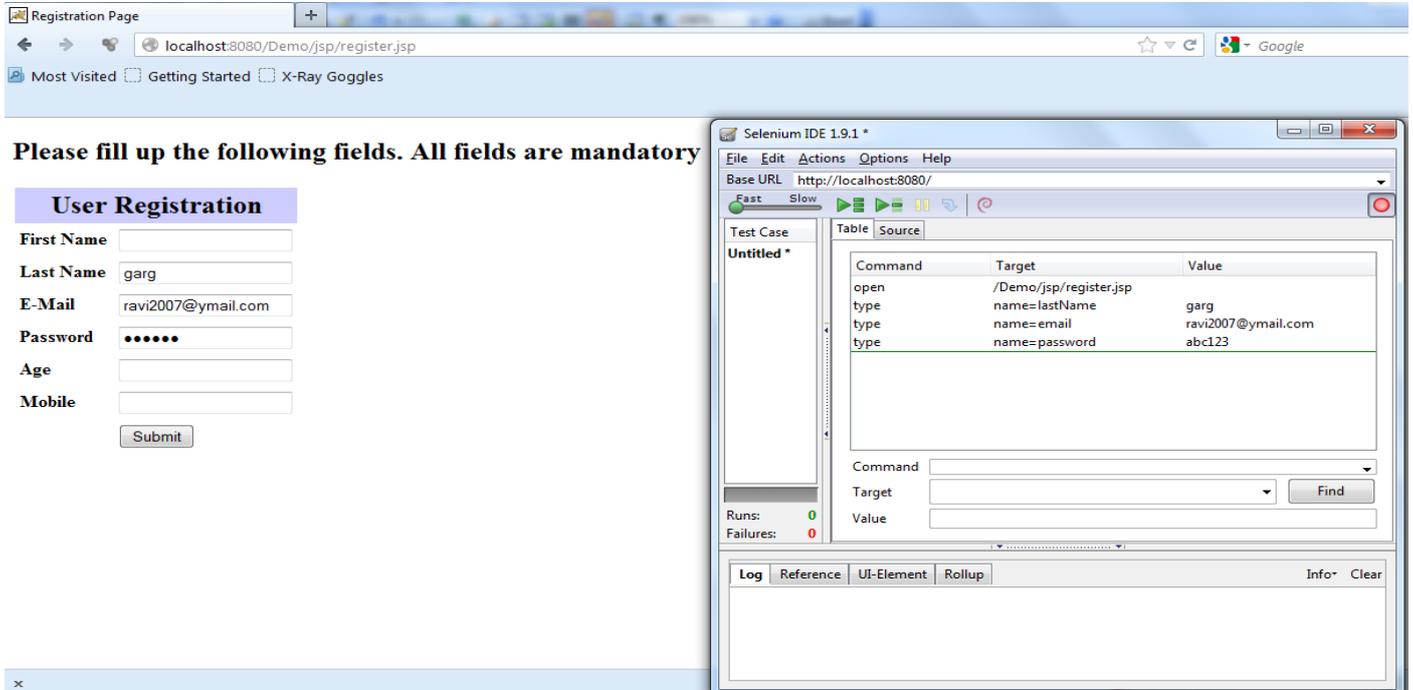
Figure 5: Screen for entering the value of field E-Mail & corresponding record in selenium.

(iii) When value of E-mail is entered in the web page, the command corresponding to that is recorded on the selenium IDE with command "type" target "name =email" and value "ravi20007@ymail.com". Type command corresponds action taken by user, means typing the data.
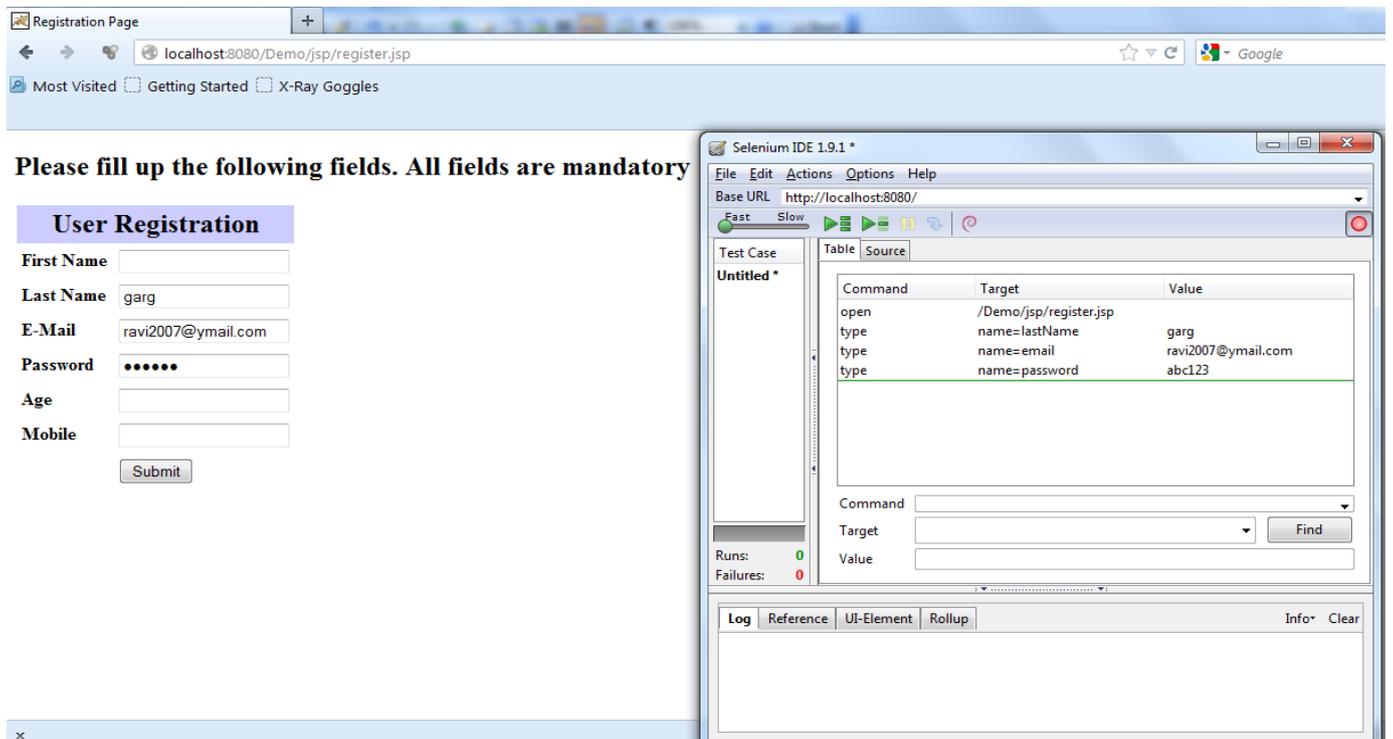
Figure 6: Screen for entering the value of field Password & corresponding record in selenium.

The value of password is entered in the web page, the command corresponding to that is recorded on the selenium IDE with command "type" target "name =password" and value "abc123". Type command corresponds action taken by user, means typing the data. "Target" on selenium screen corresponds to field whose value is to be executed during execution of the test case.
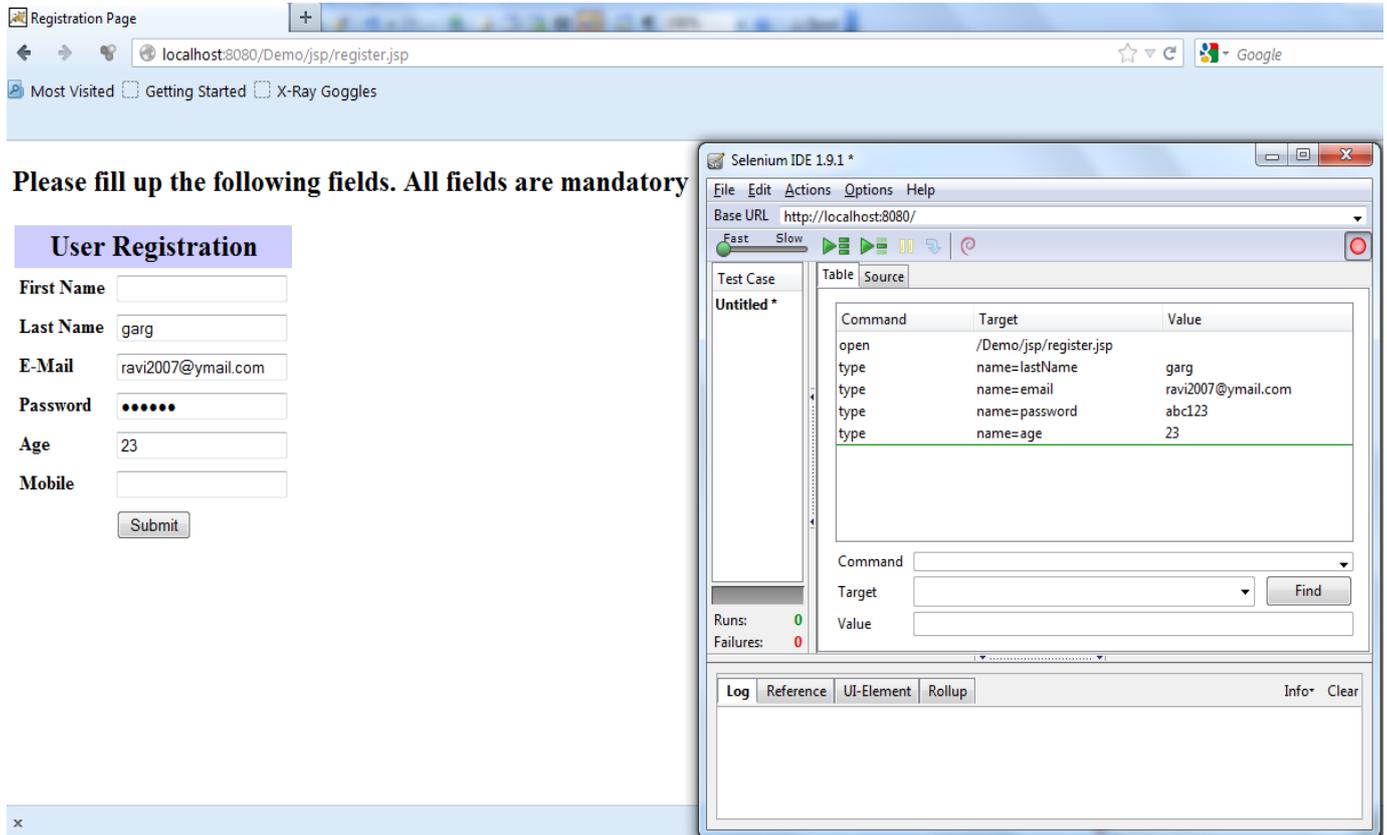
Figure 7: Screen for entering the value of field Age & corresponding record in selenium.

The value of "age" is entered in the web page, the command corresponding to that is recorded on the selenium IDE with command "type" target "name =age" and value "23". Type command corresponds action taken by user, means typing the data. "Target" on selenium screen corresponds to field whose value is to be executed during execution of the test case. "value" is the expected output which should match with the actual value during the run of the test case.
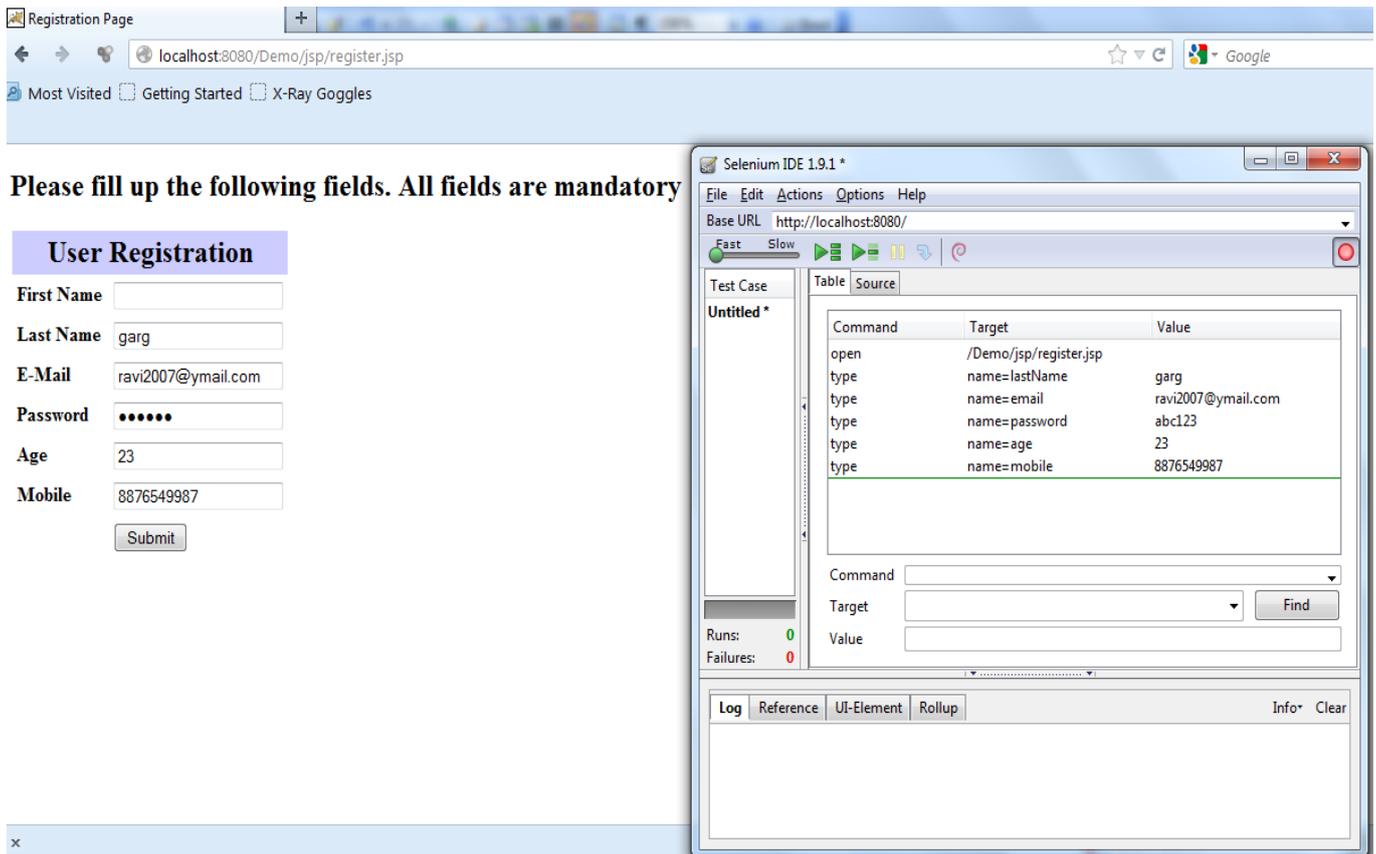
Figure 8:Screen for entering the value of field Mobile & corresponding record in selenium.
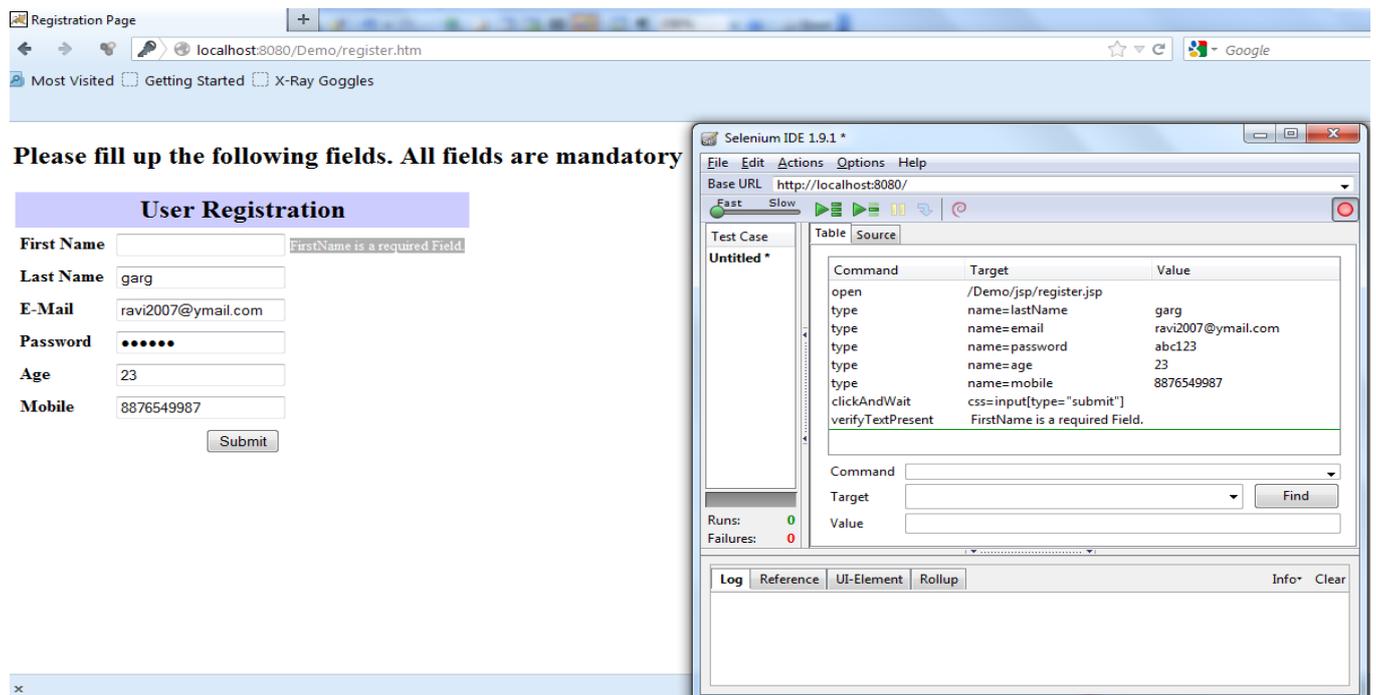
Figure 9:Screen for entering the command " verify text present" in selenium .

After entering the data , a screen will appear as shown in figure 9. then user has to stop recording by clicking on red button & to enter a command " verify text present" & target value "FirstName is a required field". This command is used to search the message as user has not entered the data for the field "first Name".

Figure 10: Screen for successful runnikng of test case no. 4 given in Table 1.

Using selenium, when we run the test case using run button, then all the values of the fields are matched with the target values, whose values are entered through TYPE command. But the values of the message will be matched through Verify text present.

CONCLUSION

Automated Software testing is the best way to increase the effectiveness, efficiency and coverage of software testing and Selenium is a framework comprises of many tools used for testing web applications. With the help of the case study, we analyze and find the testing of a web application using automation testing tool "Selenium IDE". Using this approach, test cases are automatically recorded in background while tester is entering the data in a web application screen and these test cases are reusable and best suited in the Regression Testing environment.

BIBLIOGRAPHY

[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