International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878, Volume-8 Issue-3, September 2019

Impact of Unit Testing in Web Automation Testing

G.Chandrasekaran, V.Neethidevan, J.Murugachandravel

Abstract: ***The main intent of performing Software testing is to see that the software system is working as per the user's requirements. Unit testing is done at the individual unit or at the components level. The main objective of this testing is to validate and ensure that each unit is performing its required function effectively. In case of automated testing, in recent days, many testing tools are available which help developers to perform unit testing in a simpler manner. This paper shows how automated unit testing is performed using different testing tools like JUnit, Selenium with web driver Java and Selenium with python. Regarding the experimental work, different applications were tested with the automated tools and results have been discussed with its code. To ensure reliability on web testing, the application has to be tested with the most suitable test case. With Selenium web driver concept, various User Interfaces are identified with xpath and CSS concept and test scripts ensure that various inputs are navigated between User Interface elements. The Automated software testing process reduces time and makes it more accurate.***

Keywords: Unit testing, Selenium testing tool, Junit, Nuint, Unit testing Framework, Web Automation.

I. INTRODUCTION TO UNIT TESTING.

In recent days, the developers have to develop each module and then test the module with different test cases to make the module free from errors. Further, they are using different automated tools that reduce the testing time. Once unit testing is performed, it reduces the number of bugs. To perform the unit testing, there is a necessity for selecting the suitable test cases. The different guidelines to be followed to perform unit testing are listed below:

- Identify a suitable tool/framework based on the given application. For example, Junit is a unit testing tool used to perform unit testing on an application developed using java programming language. Eclipse is a Junit testing framework that provides necessary helps to the developer by writing minimal set of Java code.
- Create any suitable test case for current unit by recognizing functionalities of each unit.
- Have a separate lab to do testing with the necessary infrastructure.
- Use the most suitable test case that can identify errors properly.
- To fix the defect, do a root cause analysis to find the main reason for error and then try to fix the error.

1.1 Selenium web driver concept

- It helps in providing a framework to perform web automation testing with different browsers like Firefox, Chrome, Opera, etc. Once the Selenium Java code is ready, testing is done with different browsers.

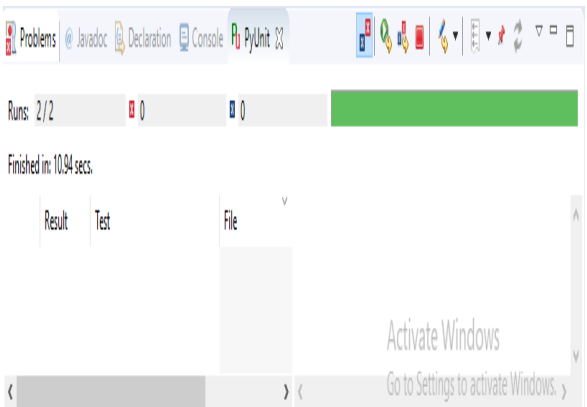
II. LITERATURE SURVEY

In [1], Andrew Patterson et al, addressed the testing problem by combining two existing unit testing frameworks JUnit and BlueJ, and they created a new testing tool that was flexible enough to use BlueJ system and Junit. In [2], Arpit Christi et al, proposed a testing tool for code coverage based testing. In [3], MagielBruntink et al, explored various factors for testing object-oriented software systems. In [4], Catherine Oriat et al, presented a tool, Jartege, useful for generation of unit tests for Java classes specified in JML (Java Modeling Language), which is a specification language for Java. In [5], ErmiraDaka et al, described about the importance of Unit testing and implemented a java calculator program on the Net beans platform and tested its main components under JUnit4 testing tool. In [6], Ravinder Kumar et al, explored the possibility of using Junit with test suites. In [7], Marcel B¨ohme and Soumya Paul explored the relative efficiencies of the random and systematic approaches to automated software testing.

III. EXPERIMENTAL WORK

Python Unit test

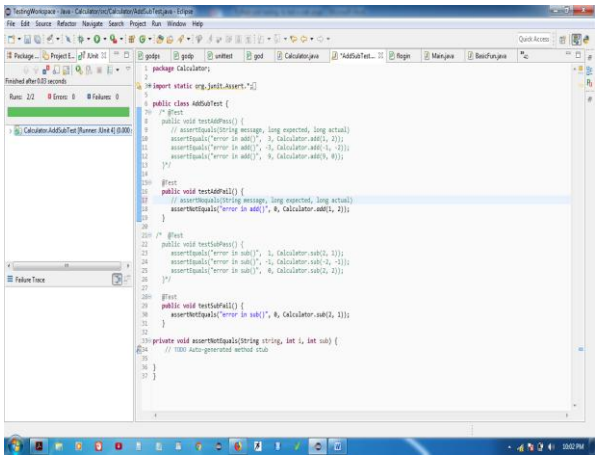
The unit test using python code is done for checking amicable numbers is done and the code is available at Appendix section.

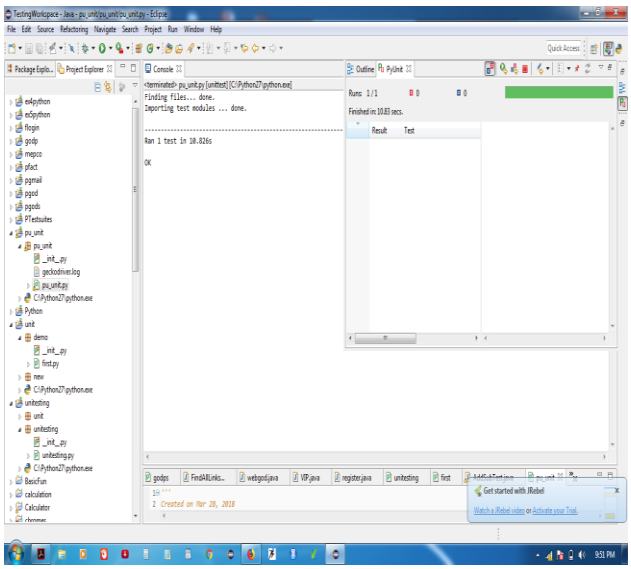


Python unit testing framework for Web testing

Manuscript published on November 30, 2019.
* Correspondence Author
G.Chandrasekaran*, MCA Department MepcoSchlenkEngg. College (Autonomous), Sivakasi, Tamil Nadu, India
V.Neethidevan, MCA Department MepcoSchlenkEngg. College (Autonomous), Sivakasi, Tamil Nadu, India
J.Murugachandravel, MCA Department MepcoSchlenkEngg. College (Autonomous), Sivakasi, Tamil Nadu, India

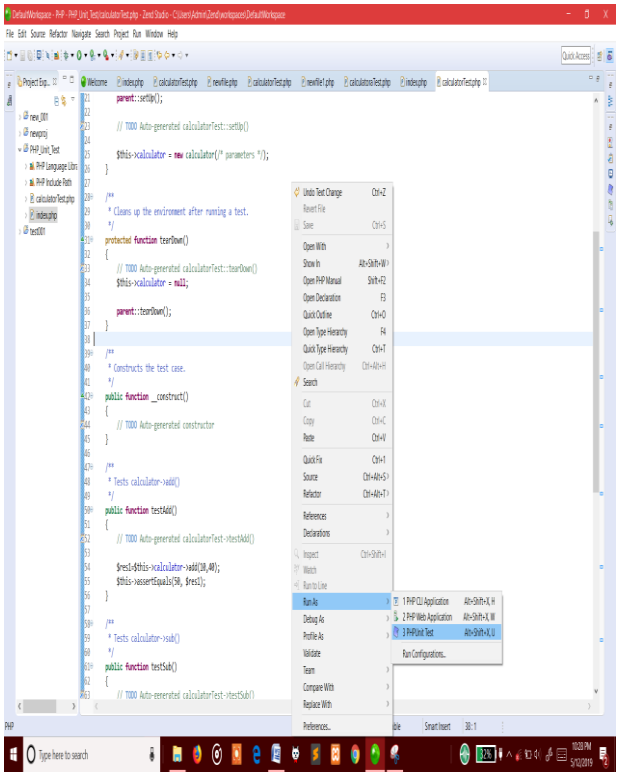
To test a login page, code is written using selenium web driver with python unit testing framework. The test is conducted on the google search engine, whether it loads the web page properly and then searches details like selenium interview questions without any problem. In our study, two pages were loaded successfully and green bar was displayed to indicate that the test was passed successfully. In addition to that, the same application was loaded with different browser and tested for cross browser testing, resolution; various text boxes and labels are loaded properly in all the browsers. Junit is a Unit testing framework for Java programs. Here unit testing is conducted using Junit for a calculator program as shown in appendix the part.





PHPUnitTest

This testing framework was used to test programs developed using PHP. It was developed under Zendstudio, a framework for PHPUnit test.



IV. CONCLUSION

Thus the unit testing was performed using Junit testing tool and Selenium python unit testing for web based application. Unit testing was also conducted for Python and PHP. The above tests were conducted using automated tools so that it could save time and perform its intended functions properly. As automated testing was performed for unit testing, it saved time and effort by the developer. More number of errors was identified at unit level. Thus, four unit testing tools were demonstrated for various languages like Java, Python, PHP and Python web testing for unit testing.

REFERENCES:

1. Patterson, A., Ko¨lling, M., & Rosenberg, J. (2003). Introducing unit testing with BlueJ. ACMSIGCSE Bulletin, 35(3), 11 – 15.
2. Arpit Christi, Autocodecovergen: Prototype Of Data Driven Unit Test Genration Tool That Guarantees 100% Code Coverage For C#, Advanced Computing: An International Journal (ACIJ), Vol.3, No.5, September 2012.
3. MagielBruntink a Arie van Deursen, An Empirical Study into Class Testability, Journal of Systems and Software 16 February 2006
4. Catherine Oriat. Jartege: a Tool for Random Generation of Unit Tests for Java Classes. 2004.
5. ErmiraDaka and Gordon FraserUniversity of SheffieldSheffield, United Kingdom, A Survey on Unit Testing Practices and Problems.
6. Ravinder Kumar, Mr. Karambir Singh, Enhancing Component Based Testing Using JUnit Tool in Net Beans Environment, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 7, July 2012.
7. Marcel B¨ohme and Soumya Paul, A Probabilistic Analysis of the Efficiency ofAutomated Software Testing, TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. ??, NO. ??, ??? 2015
8. http://softwaretestingfundamentals.com/unit-testing/

AUTHORS PROFILE

The author Dr. G. Chandrasekaran is working as a Senior Professor and Director in the Department of Computer Applications. He has more than 30 years of Teaching experience at Mepco Schlenk Engg college, Sivakasi and 4 years Research Experience at IIT, Delhi. So far, he has published 80 papers in International/National journals and conferences. He has organized 70 courses both National level and State level in the form of FDP, National and International Conferences, Workshop and Seminars. He has also participated in 80 different courses like Workshops, Summer and Winter courses conducted at National and State levels. His areas of interest include Software Engineering, Digital Image Processing and Linguistic Computing.

He is a recipient of the following awards.

i) Dr. A.P.J. Abdul Kalam Award for Life Time Contribution in Teaching in 2016.
ii) Dr. A.P.J. Abdul Kalam Award for Scientific Excellence in 2015.
iii) ISTE Periyar Award for Best Engineering College Teacher in 2007.

The author Mr. V. Neethidevan is working as Assistant Professor (SLG) in MCA Department of Mepco Schlenk Engineering College (Autonomous), Sivakasi with more than 20 years of teaching experience. He attended more than 50 different courses both at National and State levels. He has also organized more than 15 different courses of the type Workshop, Seminar and FDP. His areas of interest include Data mining and Digital Image Processing. He completed MCA from Bharathidasan University, Trichy in 1996 and ME (CSE) from Anna University, Chennai in 2007.

The author Mr. J. Murugachandravel is working as Assistant Professor (SL.G) in MCA Department of Mepco Schlenk Engineering College (Autonomous), Sivakasi with more than 18 years of teaching experience. He has attended more than 50 different courses both at National and State levels. He has also organized more than 15 different courses of the type Workshop, Seminar and FDP. His areas of interest include Databases and Digital Image Processing. He completed MCA from Madurai Kamaraj University, Madurai in 1997 and ME (CSE) from Anna University, Chennai in 2008.