IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

UI AUTOMATION FOR IVR SYSTEM

Prof. Vina M Lomte, Praveen Kumar Sahu, Mukesh Kumar Verma, Amarendra Kumar, Prashant Balaji Birajdar
Department of Computer Engineering, RMD Sinhgad School of Engg, Pune, India

ABSTRACT

As we know Software testing is quite a tedious and extravagant work ,so there is a need of overcoming this problem by my making some tool to reduce human involvement and reduce time to perform nearly similar tasks. UI Automation tool will automatically run the test cases using the records or data present in XML file.

The growth of the tool involves the usage of Selenium IDE along with Visual Studio and Languages such as C#,Java. Basically the tool is used to automate testing of web based application. Initially we need to insert the Test data in the XML file ,then the data is accordingly utilized to test the application to generate the success or failure of particular test case to generate Error or Success report.

KEYWORDS: UI Automation ,Visual Studio,Error / Success report,Test data,Selenium IDE.

INTRODUCTION

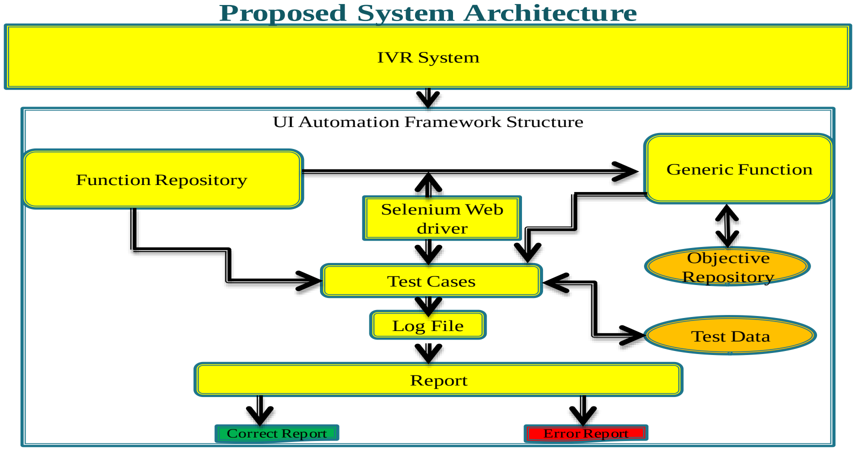
We are familiar with the crave of the current growing technology and the agility with which it is growing of the newly researched products . Softwares or the tools are very essential to handle these products efficiently ,so as to fulfill the requirements of the clients or the customers, also to do the same we need to audit and test the tool on the regular basis either by providing the application continuance support or by periodically examining. Due to advancement of the current technology there are various web applications made for the same and fortunately the mob is taking interest and utilizing the applications very largely. Hence testing is very much required ahead of flowing the software tool in to the merchandise or precisely the market.

MATERIALS AND METHODS

Tools Used

Selenium is an open source automation testing tool for web based applications. It runs directly on browser and supports almost all available browsers such as Firefox, chrome, IE, Opera, Safari etc. It runs on all platforms such as Windows, Linux and Macintosh. It‟s a very useful tool for System functional testing and browser compatibility sting.

Proposed System Architecture



Algorithm Used

```
Step1. Start
Step2. Build Project
Step3. Click on debug using N-unit tool
Step4. Select test cases in N-unit tool
Step5. Click on debug run in N-unit tool
Step6. Create empty report folder(in html format)
Step7. Read and save test data and objects into hashtable
Step8. Execute test cases
        If (PASS)
          then
               goto (generate test case level report)
          else(FAIL)
               then take screenshot and goto(generate test case level report)
Step9. Test case available to execute
          If(yes)
          then
                goto(execute test cases)
          else(no)
                generate final report
Step10. Exit
```

TEST PROCEDURE

The user first needs to build the project. Then using N-Unit he will be able to select test cases. For this user needs to click on "Debug using N-Unit" . This would open the N-Unit window into which user can check the check box of test cases he wishes to execute. And then just click on run button in N-Unit. After the execution is over the user can see the reports generated in the Reports folder.

ABOUT THE FRAMEWORK

Steps

Getting Code

- Create folder.
- Right click on folder then on SVN and then Checkout.
- Paste URL in then dialog box opened.
- Click on Ok button .

Starting Framework

- Open the folder created, double click on .sln file.
- User/Tester can view all the files in solution explorer of Visual Studio.

CONFIGURING DATA

- User/Tester can set configuration data like URL of application, browser to be used for testing and the timeout time.
- Configuration data is set in GlobalConfig.xml file.
- The GlobalConfig.xml file reside in the "TestData" folder.

TEST DATA

- User/Tester can set test data required to execute the test cases.
- The test data files reside in TestData folder.
- The name of the file is same as the test case file name (<TestCaseName>.csv).
- The first line of the test data file is the title of test data and the second line is the actual test data.

EXECUTING TEST CASES

- To execute the test case(s) first build the project.
- Click on BUILD and then BuildSolution
- Then click on TOOLS followed by N-Unit and then"Debug N-Unit GUI"
- Then select test case(s) in N-Unit which are to be executed.

EXECUTION RESULT

- In the N-Unit window user/tester can see the result of test cases. If the test case result is PASS it is indicated with GREEN color. If the test case result is FAIL it is indicated with RED color. Details of exceptions are also given.

- Execution results contains script execution logs which can be used to keep track of how the test execution behaves.
- **Customizes csv result log:** The test log in .csv file will contain entries in following sequence- Test case name, step number, test result, date and time of execution.

RESULTS AND DISCUSSION

Test Procedure

The user first needs to build the project. Then using N-Unit he will be able to select test cases. For this user needs to click on "Debug using N-Unit" . This would open the N-Unit window into which user can check the check box of test cases he wishes to execute. And then just click on run button in N-Unit. After the execution is over the user can see the reports generated in the Reports folder.

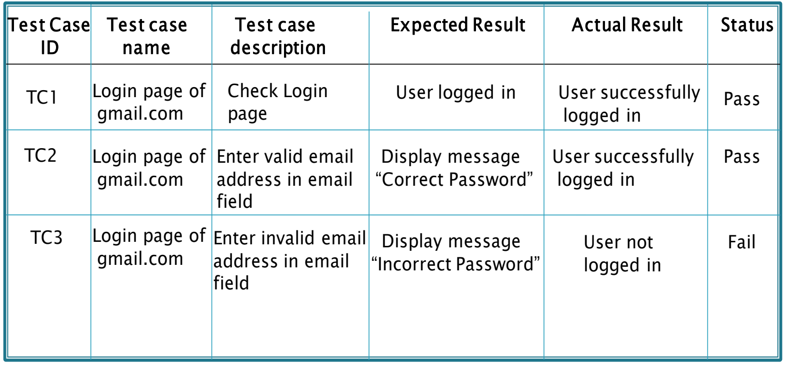
Test Cases

The test cases consists of total six columns as name of the test case ID,test case name,test case description,expected

Finally the Report

- Report generated is in html format.
- The report consists of : -
    i. Total number of test cases executed.
    ii. Total number of test cases passed.
    iii. Total number of test cases fail.
    iv. Host name/PC name.
    v. Start date and time of execution.
    vi. End date and time of execution.
    vii. Test case name with result.
    viii. Test case steps details with result.
    ix. Screen shots if the test case result is fail

result,actual result and status of the project locker if the test result is failed and that particular bug is locked. The test cases also consist of different tests that are browser specific. For these the column of browser is added to the test case sheet. The test cases are written according to test test plan. Below is an example



Test Case ID	Test case name	Test case description	Expected Result	Actual Result	Status
TC1	Login page of gmail.com	Check Login page	User logged in	User successfully logged in	Pass
TC2	Login page of gmail.com	Enter valid email address in email field	Display message "Correct Password"	User successfully logged in	Pass
TC3	Login page of gmail.com	Enter invalid email address in email field	Display message "Incorrect Password"	User not logged in	Fail

CONCLUSION

As software testing is required after development of any application , and more number of people are required if the testing is manual.So if we automate the manual testing it will reduce time , less human resource will be required so software developed will be more efficient.

REFERENCES

1. http://www.faqs.org/qa/qa-6667.html
2. Wikipedia: Test Automation http://en.wikipedia.org/wiki/Test_automation
3. N. Tracey, J. Clark, K. Mander and J. McDermid, Automated test-data generation for exceptionconditions, Software Practice and Experience, vol.30,no.1,pp.61- 79, 2000.

4. A. M. Memon and Q. Xie, Studying the fault-detection effectiveness of GUI test cases for rapidlyevolving software, IEEE Trans. on Software Engineering, vol.31, no.10, pp.884-896, 2005.
5. Lessons Learned in Software Testing, by C. Kaner, J. Bach, and B. Pettichord
6. Effective Software Testing, by E. Dustin

AUTHOR BIBLIOGRAPHY

	Prof Vina M Lomte is an Asst. Professor under Department Of Computer Engineering. She is having 10+ years‟ experience in the field of teaching as well as research. Her Research interests include domains like Web Security and Software Engineering/Testing.
	Praveen Kumar Sahu is a student pursuing his B.E Degree under Department Of Computer Engineering from RMDSSOE College, University Of Pune . He is currently working on the Mathematical Model and the transitions involved in the model of the Framework
	Mukesh Kumar Verma is a student pursuing his B.E Degree under Department Of Computer Engineering from RMDSSOE College, University Of Pune. He is presently involved in the system developing activities for developing algorithm for Framework. Also he is actively working on languages such as C # ,XML,Scripts.
	Amarendra Kumar is a student pursuing his B.E Degree under Department Of Computer Engineering from RMDSSOE College, University Of Pune. He is currently involved in the development of different Automation Frameworks and Code Coverage Mechanism of the system. He is also involved in the development of testing and automation scripts using languages such as C #.
	Prashant Balaji Birajdar is a student pursuing his B.E Degree under Department Of Computer Engineering from RMDSSOE College, University Of Pune. He is presently working on Mathematical Module of the Framework.