Analysis of Automation and Manual Testing Using Software Testing Tool

Priyanka Rathi  
Student (MCA),  
Rukmini Devi Institute of  
Advanced Studies, Rohini, New Delhi

Vipul Mehra  
Test Engineer  
IBM INDIA PVT LTD

ABSTRACT

Testing is a major activity in software development process to find the defect in the software. It means testing in IT world is used to show how we can test and where our application or software is lacking. We have Testing can be conducted manually as well as automated. This paper presents the concept of automation and manual testing and problem with manual testing and benefit of automatic testing. This paper is also show the two forms of testing which is manual and automation (we use QTP ) and how they differ from each other and why we use both. This paper also show data driven testing. The main objective of this research paper is to focus on, effectiveness and importance of automation testing.

KEYWORDS
Test Cases (TC), QTP, Software Development Life Cycle, Software Testing Life Cycle

Introduction

In recent years, software testing is becoming more popular and important in the software development industry. Testing is the critical element of software quality assurance and represents the ultimate review of specification, design, and coding. Software can be tested either manually or automatically. To create test cases manually and execute them without any tool support is known as manual testing . Manual software testing is performed by a human sitting in front of a computer carefully going through application screens, trying various usage and input combinations, comparing the results to the expected behaviour and recording their observations . Automation Testing means using an automation tool to execute test suite. Goal of automation is to reduce number of test cases to be run manually and not eliminate manual testing all together . Some automation tools are: Winrunner, Loadrunner, QTP 9.2,9.5,10.00 and 11.00 etc.

TESTING

Manual Testing is the form of Testing in which we test the given product,Software or application to test and we test it Manually. It is the process of evaluating the system or system components manually. Manual means the amount of men work will be more but in Automation we will automate a given product by writing scripts in different scripting language(QTP is used ) .The men work will be less and it will perform faster and accurate results.

<table>
<thead>
<tr>
<th>MANUAL</th>
<th>AUTOMATION(QTP)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. More Testers are required</td>
<td>1. Few Testers are required</td>
</tr>
<tr>
<td>2. Takes lot of time to execute a test case</td>
<td>2. Saves Time as compared to manual</td>
</tr>
<tr>
<td>3. No licence</td>
<td>3. Usually have Licence</td>
</tr>
</tbody>
</table>
1. Mode of Testing

Based on test execution, testing can be classified into two categories.

a. Manual Testing
b. Automation testing

1. Manual Testing - Manual testing is a testing technique where the test engineer prepare test case manually and execute them to identify defect in the software. It is most rigorous and old method of software testing. 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. Repetitive manual testing can be difficult to perform on large software applications or applications having very large dataset coverage.

Problems with manual testing

Time consuming - Since test cases are executed by human resources so it is very slow and tedious.

Huge investment in human resources - As test cases need to be executed manually so more testers are required in manual testing.

Less reliable - Manual testing is less reliable as tests may not be performed with precision each time because of human errors.

Non-programmable - No programming can be done to write sophisticated tests which fetch hidden information. Manual Testing can become boring and hence error prone

2. Automation Testing

Automating software testing involves developing test scripts using scripting languages such as Python, JavaScript, so that test cases can be executed by computers with minimal human intervention and attention. Test design and development together can be automated to reduce human effort and save cost. The automation software can also enter test data into the system under test, compare expected and actual results and generate detailed test reports. Test Automation demands considerable investments of money and resources. Successive development cycles will require execution of same test suite repeatedly. Using a test automation tool it’s possible to record this test suite and re-play it as required. Once the test suite is automated, no human intervention is required. Goal of automation is to reduce number of test cases to be run manually and not eliminate manual testing all together.

Benefits of automation testing

Fast- It is faster than the manual testing.

Cost Effective- Test cases are executed by using automation tool so less tester are required in automation testing

Repeatable- The same test case (record and replay) can be re-executed using testing tools [9].

Reusable- Test suits can be re-used on different versions of the software.

Programmable- Testers can program sophisticated tests that bring hidden information.

Comprehensive- Testers can build test suites of tests that cover every feature in software application.

More reliable- Automation tests perform precisely same operation each time they are run.

Test Coverage- Wider test coverage of application features.

TOOLS FOR AUTOMATED SOFTWARE TESTING:

For many test managers, the decision of which testing tools to use can cause confusion. The first decision to make is which category of tool to use— one that tests specific units of code before the application is fully combined,
one that tests how well the code is working as envisioned, or one that tests how well the application performs under stress. And once that decision is made, the team must wade through a variety of choices in each category to determine which tool best meets its needs. Evaluating your needs will narrow down your short list very rapidly. There are many options in market to choose. Depending on what sort of test it will be used for the choice may become easy. The following general categories categorize the tests and tools which makes it easy to choose

1. Developer oriented tools
2. Functional testing tools
3. Load testing tools
4. Performance monitoring and maintenance tools

DEVELOPER ORIENTED TOOLS:
Tools in this category, also called component testing tools or unit testing tools, test individual software components or groups of related components, helping isolate and rectify problems as early in the development process as possible. Areas of testing include memory analysis, function analysis. The important tools in this area are

- DevPartner Studio from Compuware Corp.
- PurifyPlus from IBM Rational.
- Optimizeit Enterprise Suite from Borland Software Corp.
- C++Test, Jtest and TEST from Parasoft.

FUNCTIONAL TESTING TOOLS:
Tools in this class help verify that applications will work as expected. It allows developers to record an existing application and modify scripts to meet changes in an upcoming release. They also provide for regression testing on the new release using the test scripts developers have captured up until that point. The important tools in this area are

- WinRunner from Mercury interactive Corp.
- Astra QuickTest Mercury interactive Corp.
- SilkTest from Segue Software Inc.
- RationalSuite TestStudio from IBM Rational.

LOAD TESTING TOOLS:
This category, also called performance or stress testing tools, tests what happens to the code as the application scales with multiple users in a simulated environment. The tools test, among other things, whether performance degrades as the load is increased. Once bottlenecks are found, these tools can determine the source of the problem and begin fixing it. The important tools in this area are

- LoadRunner from MercuryInteractive.
- SilkPerformer from Segue Software Inc.

PERFORMANCE MONITORING AND MAINTENANCE TOOLS:
All the tools in this category are used after the application is already in production by examining how the application is working in a real-world environment. In essence, these tools monitor the production environment to ensure that all requirements and defined thresholds are continually being met. The important tools in this area are

- Vantage from Compuware.
- VTuneAnalyzer from Intel Corp.

Data Driven Testing:
The execution of a test with multiple sets of test data is called Data Driven testing. It is also called as Re-Testing. There are different ways of doing data driven testing:

- Dynamic Test data submission
- From Front-end objects
- From Flats files
- From Excel Sheet
- From XML files
- From Database
DYNAMIC TEST DATA SUBMISSION
Sometimes, you may need to submit test data dynamically when the test scripts get executed.
1. When the scripts executes, you get an Input Box called login
2. When you enter the Agent Name
3. Inserts the Agent Name in the Flight Reservation application’s “Agent Name:” edit box.
Here, you dynamically provide input test data during the test execution and hence its name.

FRONT END OBJECTS
Sometimes, you may need to fetch values from an application (Front-end object) and based on the type of the values you fetch, you may need to insert the values in some other objects in the same/different application and execute the scripts. This method of driving the test using front-end objects is called Data Driven testing using front end objects.

FLAT FILES
Sometimes, you may need to fetch data/values from flat files like Notepad and drive your script execution.

EXCEL SHEET
You can either insert data in the Data Table available for the script and use the values using Data Table. Value("<parameter_name>",sheetid) in your script or you can import an already existing excel sheet.
Importing an already existing excel sheet with test data:
1. Assume there is a test data xls.
2. You can import the xls into your script and use the data.
3. You can directly import the data sheet manually to the Data table.
4. Select “ok” when you get the pop up
5. Select the appropriate sheet,
   You now get Data Table filled in with the available data
6. You cant execute your script using all the data rows.
7. Navigation File->Settings->run tab->Select the option “run one iteration only”
   Instead of using Data Table Of Qtp, you can also directly use VBScript program to fetch values from the input excel.

The following program fetches the login test data from excel sheet and executes the login functionality of the Flight Reservation application.
```vbnet
Set oExcel = CreateObject("Excel.Application")
strExcelPath="d:\automation\testdata\fralogin.xls"
set oWorkBook=oExcel.WorkBook.Open(strExcelpath)
set oWorkSheet=oExcel.WorkBook.WorkSheets(“sheet1”)
oWorkSheet.UsedRange.Select
intRowsCount=oWorkSheet.UsedRange.Rows.Count
for i=2 To intRowsCount
    SystemUtil.Run"c:\programfiles\hp\quick test professional\samples\flight\app\flight4a.exe"
    strAgentName=oWorkSheet.Cells(i,1).value
    strPassword=oWorksheet.cells(i,2).value
    Dialog(“Login”).WinEdit(“Agent Name:”).Set Trim(strAgentName)
    Dialog(“Login”).WinEdit(“Password:”).Set Trim(strPassword)
    Dialog(“Login”).WinButton(“ok”).click
    If Window(“Flight Reservation”).WinButton(“FLIGHT”).Exit(20)Then
        Window(“Flight Reservation”).Close
    Else
        Reporter.ReportEvent micFail,”Step 02:Iteration”& i-1,”Login Failed”
    End If
Next
```

Reporter.ReportEvent micFail,"Step 02:Iteration"& _i-1,"Login Not Successful with the Credentials"
&"AgentName/Password:",&StrAgentName&/&strPassword
SystemUtil.CloseProcessByName “flight4a.exe”
End If
Next
oExcel.Quit
You can also fetch the data from database instead of excel sheet, if your test data resides in the data base

**Navigation**

1. Select the “specify SQL statement manually” option and click on “Next” button.
2. Click on Create-> and select the appropriate data source.
3. Write the appropriate SQL Query and click on the “Finish” button.
4. You now get the data in the Data Table of QTP.

**CONCLUSION**

Manual testing is time consuming, tedious and requires heavy investment in human resources. Automation tools enable us to record the test suite and re-play it if required. Once the test suite is automated, no human intervention is required. In automation testing the initial investments are bigger than manual testing and you cannot automate everything but automatable test cases, determine which ones (manual or automated) would provide the biggest return on investment. Metrics are an important to analyze the quality, and progress of an automated software testing and manual testing effort. The test metrics provides the visibility into the readiness of the product and give clear measurement of the quality and completeness of the product. This paper discussed some manual testing metrics and automation metrics like test execution, test case productivity, defect rejection metrics test coverage metrics to evaluate the performance of both types of testing.

**REFERENCES**