REVIEW OF AUTOMATION SOFTWARE TESTING (AST) AND IMPLEMENTATION OF RFT AUTOMATION TESTING TOOL FOR AUTOMATION SOFTWARE TESTING

Shilpa Mahajan
Assistant Professor, Computer Science department, Dronacharya College of Engineering, Gurgaon, Haryana. E-mail: Shilpa.mahajan284@gmail.com

Abstract: Software Testing is of mainly two types: Manual Testing and Automated Testing. Manual testing uses the knowledge of the testing engineer to target testing to the parts of the system that are assumed to be more error-prone. Software quality is improved by the use of automated testing. Automated testing cannot replace manual testing but it is used to detect that faults/defects which manual testing alone cannot find. RFT-Rational Functional Tester tool is used to test the functionality of software. It provides testers with automated testing capabilities for functional testing, regression testing, and GUI testing and data-driven testing. In this paper, RFT is used to test the functionality of blood-bank software. To do this, first test cases are generated manually. After this, scripting is done using java script.

Index Terms: Software testing, AST, RFT

I. INTRODUCTION

I.1. Software Engineering

It is the application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software. Software engineering can be divided into ten sub disciplines. Various disciplines of Software Engineering:

• Software development
• Software testing
• Software requirements
• Software Design
• Software Maintenance
• SCM
• Software Quality

I.2. Software Testing

Software testing is the process of executing a program or system with the intent of finding errors. Software Testing can also be stated as the process of validating and verifying that a software program/application/product:

1. meets the business and technical requirements that guided its design and development;
2. works as expected; and
3. Can be implemented with the same characteristics.

I.3. Software Testing Cycle

Testing is usually performed for the following purposes:

• To improve quality.
• For Verification & Validation (V&V)
• For reliability estimation

I.4. Types of Software Testing

1. Manual Testing
2. Automation Testing
Manual Testing: It is the process of manually testing software for defects. It requires a tester to play the role of an end user, and use most of all features of the application to ensure correct behavior. To ensure completeness of testing, the tester often follows a written test plan that leads them through a set of important test cases. Manual testing has certain limitations which include:

- It is very time consuming.
- It is error-prone because test engineers become bored when testing has to be done repeatedly.
- To do performance testing, many resources are required, both computer and people.
- Regression testing has to be done to ensure that changes in one portion of the software have no ill effects on the other portions of the software. The entire testing process needs to be repeated whenever a change is made to any portion of the software.

Automation Testing: Although manual tests may find many defects in a software application, it is laborious and time consuming process. In addition it may not be effective in finding certain classes of defects. Test automation is a process of writing a computer program to do testing that would otherwise need to be done manually. Once tests have been automated, they can be run quickly.

II. AUTOMATION SOFTWARE TESTING (AST)

II.1. Automation Testing/Test Automation

It is the use of software to control the execution of tests, the comparison of actual outcomes to predicted outcomes, the setting up of the preconditions and other test control and test reporting functions. It automates not only test case execution, but also test case generation and test result verification.

Often the question is asked, “How is manual software testing different from AST”? The answer is that AST:

- Enhances manual testing efforts by focusing on automating tests that manual testing can hardly accomplish.
- Is software development.
- Does not replace the need for manual testers’ analytical skills, test strategy know-how, and understanding of testing techniques. This manual tester expertise serves as the blueprint for AST.
- Can’t be clearly separated from manual testing; instead, both AST and manual testing are intertwined and complement each other.

There are two general approaches to test automation:

- Code-driven Testing: The public interfaces to classes, modules, or libraries are tested with a variety of input arguments to validate that the results that are returned are correct. It is a key feature of agile software development, where it is known as test driven development (TDD). Unit tests are written to define the functionality before the code is written. Only when all tests pass is the code considered complete.
- Graphical User Interface Testing: A testing framework generates user interface events such as keystrokes and mouse clicks, and observes the changes that result in the user interface, to validate that the observable behavior of the program is correct.

![Automated Software Testing Process](image)

Figure 2.1: Automated Software Testing Process Realize

II.2. Various Automation Tools Available for Test Automation

- WinRunner
- LoadRunner
- QTP-quick test professional
- RFT-rational functional tester
- Test Parser
• Test Director
• Selenium
• Apodora
• Silk Test
• Rational Robot

AST is a significant area of interest in current testing research, with the aim to improve the degree of automation, either by developing advanced techniques for generating the test inputs, or by finding support procedures to automate the testing process.

II.3. ADVANTAGES OF AST
• Accuracy and completeness of performance.
• Accuracy of results log and summary reports.
• Comprehensiveness of information.
• Few manpower resources required for performing of tests.
• Shorter duration of testing.
• Performance of complete regression tests.
• Performance of test classes beyond the scope of manual testing.

II.4. DISADVANTAGES OF AST
• High investments required in package purchasing and training.
• High package development investment costs.
• High manpower requirements for test preparation.
• Considerable testing areas left uncovered.

III. LITERATURE SURVEY
Software can be tested either manually or automatically [8]. The two approaches are complementary: automated testing can perform a large number of tests in little time, whereas manual testing uses the knowledge of the testing engineer to target testing to the parts of the system that are assumed to be more error-prone. Auto Test [6] is a testing tool that provides a “best of both worlds” strategy: it integrates developers’ test cases into an automated process of systematic contract-driven testing. This allows it to combine the benefits of both approaches while keeping a simple interface, and to treat the two types of tests in a unified fashion: evaluation of the result is the same, coverage measures are added up, and both types of tests can be saved in the same format.

III.1. AUTO TEST ARCHITECTURE [6]
It is a framework for fully automated software testing. It allows for arbitrary testing strategies to be plugged in and is not hard coded to a certain testing strategy. The pluggable testing strategy is only concerned with determining exactly how and with what inputs the system under test should be invoked. The actual execution is a task of the framework. The figure is shown as:

![Figure 3.1: Auto Test Architecture](image)

A case study [1] was conducted in five organizations that develop and test technical software for the automation or telecommunication domains. The data was collected from interviews with managers, testers, and developers in each organization. Following are the observations:

1. Automated software testing may reduce costs and improve quality because of testing in less time, but it causes new costs in, for example, implementation, maintenance, and training.
2. Generic and independent products facilitate and customized and complex products hinder testing automation.
3. Low human involvement facilitates and high hinders testing automation.
4. High reusability facilitates and low reusability hinders testing automation.
5. Products based on a uniform product kernel or variants of a product family facilitate and dissimilar products hinder testing automation.

A test automation solution on GUI functional test [5] is proposed. This solution integrates test case generation and selection, test case execution, and test reporting to facilitate testing. It introduces the concept of test driver which is designed to take over the communication between test cases and the execution engine.

A new test automation solution on GUI functional test is proposed. The idea is to put test cases generation and selection, test execution and test reporting together. A test driver is introduced to interpret test cases into class objects and route them into test engine which will then execute them automatically.

Overview of functional test automation solution:

![Diagram of Test Automation Solution]

Figure 3.2: Functional Test Automation Solutions

There are different layers in the solution. At the top is test case management dashboard. It is used for test case design and test scope configuration. Once test cases are identified to be automated, a device named Test Driver starts its work to translate test cases into executable scripts such as java codes. Then test driver routes the executable scripts into an execution engine for running. The execution engine receives scripts from the driver and automatically executes the tests. A third work is to enhance the reporting engine to support customizations to meet requirements from different users.

As a summary, Figure 3.3 describes the vision of our test automation solution in near future. Automated test data generation technology is a hot researching area in automated software test. Some kinds of new concepts such as base node, control node, definition node and definition related control node set are proposed, and in addition, a solution to obtain the definition related control node set is designed. on the basis of combining both the method of linear approximation and DUC expression, an innovative algorithm based on reversed binary tree [7] for automated test data generation is proposed, which can automatically find out all of the feasible paths in the program from the source node to the base node, and can automatically generate the test data for each founded feasible path also. Thereafter, the automated test data generation problem in the software tests has been solved finally.

![Diagram of Automated Software Testing Model]

Figure 3.3: Automated Software Testing Model

Using of CI (continuous integration) [2] and various software testing techniques to achieve a robust C4ISR (command, control, communication, computers, intelligence, surveillance, and reconnaissance) multi-platform system. Designed and implemented an XML driven automated test framework that enabled developing numerous high quality tests rapidly. Since the nature of C4ISR project enforces to support multiple platforms (windows, Linux, Sun Solaris, etc.) and multi-language (C++, and Java) code development, the complexity of product is significantly increased.

IV. AUTOMATION SOFTWARE TESTING TOOLS

VI.1: Automation Testing Tool to be Used for Testing are:

- RFT: Rational Functional Tester
- RFT: Rational Functional Tester is an automated functional testing and regression testing tool. It provides testers with automated testing capabilities for functional testing, regression testing, GUI testing and data-driven testing.

It was formerly called RobotJ and XDE Tester. RFT uses Java and VB.NET as its scripting languages. There is no separate IDE for RFT, for VB.Net it is integrated with Visual Studio and for Java it is integrated with Eclipse. It is primarily used by Software Quality Assurance teams to perform automated regression testing.
V. RESULTS AFTER IMPLEMENTING RFT TOOL FOR AST

V.1. Take a Software on which Automation Testing Should be Applied
- Software - Blood bank software is taken.
- VB is used as front end and MS-Access is used as back-end.
- So to run this project, VB is installed.
- Software is shown as:

![Logon Screen of Blood Bank Software](image1)

V.2. Generate test cases
- Manually generate test cases of blood-bank software.
- Test case can be:
  1. A Condition
  2. Data requirements
  3. Pre-condition
  4. Steps
  5. An Input
  6. An Expected output

![First Screen After Login](image2)

V.3: Install Automation Testing Tools which can be used for Testing the Software
- RFT: Rational functional tester tool is installed successfully.

![Test Cases of Logon Screen of Blood Bank Software](image3)

![List of all Scripts to Test Software Using RFT](image4)

![Input Box of Login_Bloodbank After Running First Script](image5)

V.4. Scripting/Coding According to the Automation Tool
- RFT-RFT Ver 8.0 is used. This tool needs 2 GB of RAM.
- Uses Javascript and VB.NET as its scripting languages.
In this paper, Javascript is used for writing script in RFT.

If data to be collected from more than two tables in the database then we require Squirrel SQL to be installed for writing scripts for those screens or menus.

To run squirrel sql we also require java to be installed.

VI. CONCLUSION

We successfully did the implementation of AST using RFT automation software testing tool. We conclude that for testing software we have to generate test cases either manually or by using softwares/tools. And those test cases which we require many times need to be automated so that manual work can be reduced to the maximum. This is also done by us. And this is the basic function of Automation testing and for this we use RFT tool.

VII. FUTURE WORK

This implementation of AST using RFT can also be performed using QTP and other automation testing tools. So next step is to test this blood bank software using QTP-quick test professional. We have to use VBscript to write scripts in this tool for testing software. Other research can be done to minimize the disadvantages of AST so that use of AST should be maximized.

REFERENCES


