INTERNATIONAL JOURNAL OF PURE AND APPLIED RESEARCH IN ENGINEERING AND TECHNOLOGY

A PATH FOR HORIZING YOUR INNOVATIVE WORK

AUTOMATION TESTING IN SOFTWARE DEVELOPEMENT

KALPESH PARMAR

Persistent Systems Limited, Nagpur, India.

Accepted Date: 27/02/2014 ; Published Date: 01/05/2014

Abstract: The need for speed is practically the mantra of the information age. Because technology is now being used as a competitive weapon on the front lines of customer interaction, delivery schedules are subject to market pressures. Late products can lose revenue, customers, and market share. But economic pressures also demand resource and cost reductions as well, leading many companies to adopt automation to reduce time to market as well as cut testing budgets. While it might be costly to be late to the market, it can be catastrophic to deliver a defective product. Software failures can cost millions or even billions, and in some cases entire companies have been lost. So if you don't have enough people or time to perform adequate testing to begin with, adding automation will not reduce software instability and errors. Since it is well documented that software errors – even a single one – can cost millions more than your entire testing budget, the first priority should be first to deliver reliable software. Once that is achieved, then focus on optimizing the time and costs.

Keywords: WSN, 4G technology, RSVP, MRSVP, SMRP, ARSVP, WLRP.

PAPER-QR CODE

Corresponding Author: MR. KALPESH PARMAR

Access Online On:

www.ijpret.com

How to Cite This Article:

Kalpesh Parmar, IJPRET, 2014; Volume 2 (9): 871-878

INTRODUCTION

It is very important to have decent quality software's. Having this means the quality should match many requirements such as keeping the easy use of GUI's, as well as containing faults and failures and many more. A lot of effort is required to keep this quality to a reasonable standard. Testing is one of the most important parts in quality assurance especially in the development stages. As the development of the program comes to an end it becomes harder to fix errors, in fact becomes harder to spot errors.

Testing should be done during the development stages and if it is not done during the development stages then it is more than likely that there will be a lot of bugs and errors. Some problems which may not have been seen during the development stages, without testing at the end, could be something like a function being used whilst the stack is empty. This could lead to a system crashing. However, if testing is done this could be spotted before proceeding to the next stage.

Humans are prone to make mistakes and so if they do everything then it may not be as efficient. Some code may be missed out with humans and so this could cause errors in the system.

Importance of Testing Tools

As the software industry grows, it becomes more and more competitive and advanced for businesses to produce such good quality software. With this also comes the reliability and deadlines which must be met. Testing plays a big part in the case of deadlines as testing can take a long time on the software. The production of the software and the quality together must be increased for businesses to produce the best possible software. Testing, therefore, has to be done throughout the process of programming the software. Manual testing takes too long and can waste a lot of time. With the aid of testing tools this can increase efficiency and get the deadlines met.

What are Testing Tools?

Testing tools are a form of automated testing. It is basically using programs to do different testing tasks i.e. doing the tests through some form of automated method. A lot of these testing tools have commonly used coding languages such as Java and C. A lot of the tools used have different options available to help with testing.

What makes automated testing so important to these successful companies?

Automated Software Testing Saves Time and Money

Software tests have to be repeated often during development cycles to ensure quality. Every time source code is modified software tests should be repeated. For each release of the software it may be tested on all supported operating systems and hardware configurations. Manually repeating these tests is costly and time consuming. Once created, automated tests can be run over and over again at no additional cost and they are much faster than manual tests. Testing Improves Accuracy

Even the most conscientious tester will make mistakes during monotonous manual testing. Automated tests perform the same steps precisely every

Increase Test Coverage

Automated software testing can increase the depth and scope of tests to help improve software quality. Lengthy tests that are often avoided during manual testing can be run unattended. Automated software testing can look inside an application and see memory contents, data tables, file contents, and internal program states to determine if the product is behaving as expected. Automated software tests can easily execute thousands of different complex test cases during every test run providing coverage that is impossible with manual tests. Testers freed from repetitive manual tests have more time to create new automated software tests and deal with complex features.

Automation Does What Manual Testing Cannot

Even the largest software departments cannot perform a controlled web application test with thousands of users. Automated testing can simulate tens, hundreds or thousands of virtual users interacting with network or web software and applications.

Automated QA Testing Helps Developers and Testers

Shared automated tests can be used by developers to catch problems quickly before sending to QA. Tests can run automatically whenever source code changes are checked in and notify the team or the developer if they fail. Features like these save developers time and increase their confidence.

Manual testing

This type includes the testing of the Software manually i.e. without using any automated tool or any script. In this type the tester takes over the role of an end user and test the Software to identify any un-expected behavior or bug. There are different stages for manual testing like unit testing, Integration testing, System testing and User Acceptance testing.

Testers use test plan, test cases or test scenarios to test the Software to ensure the completeness of testing.

Automation testing

Automation testing which is also known as Test Automation is when the tester writes scripts and uses software to test the software. This process involves automation of a manual process. Automation Testing is used to re-run the test scenarios that were performed manually, quickly and repeatedly.

Apart from regression testing, Automation testing is also used to test the application from load, performance and stress point of view. It increases the test coverage; improve accuracy, saves time and money in comparison to manual testing.

What to automate?

It is not possible to automate everything in the Software; however the areas at which user can make transactions such as login form or registration forms etc., any area where large amount of users can access the Software simultaneously should be automated.

When to automate?

Test Automation should be uses by considering the following for the Software:

- Large and critical projects.
- Projects that require testing the same areas frequently.
- Requirements not changing frequently.
- Accessing the application for load and performance with many virtual users.
- Stable Software with respect to manual testing.
- Availability of time.

How to automate?

Automation is done by using a supportive computer language like vb scripting and an automated software application. There are a lot of tools available which can be used to write automation scripts. Before mentioning the tools lets identify the process which can be used to automate the testing:

- Identifying areas within a software for automation.
- Selection of appropriate tool for Test automation.

- Writing Test scripts.
- Development of Test suits.
- Execution of scripts.
- Create result reports.
- Identify any potential bug or performance issue.

Software testing tools

Following are the tools which can be used for Automation testing:

- HP Quick Test Professional
- Selenium
- IBM Rational Functional Tester
- SilkTest
- TestComplete
- Testing Anywhere
- WinRunner
- LaodRunner

Testing Methods

Black Box Testing

The technique of testing without having any knowledge of the interior workings of the application is Black Box testing. When performing a black box test, a tester will interact with the system's user interface by providing inputs and examining outputs without knowing how and where the inputs are worked upon.

White Box Testing

White box testing is the detailed investigation of internal logic and structure of the code. White box testing is also called glass testing or open box testing. In order to perform white box testing on an application, the tester needs to possess knowledge of the internal working of the code.

The tester needs to have a look inside the source code and find out which unit/chunk of the code is behaving inappropriately.

Grey Box Testing

Grey Box testing is a technique to test the application with limited knowledge of the internal workings of an application. In software testing, the term the more you know the better carries a lot of weight when testing an application.

Mastering the domain of a system always gives the tester an edge over someone with limited domain knowledge. Unlike black box testing, where the tester only tests the application's user interface, in grey box testing, the tester has access to design documents and the database. Having this knowledge, the tester is able to better prepare test data and test scenarios when making the test plan

Automation Process

Following steps are followed in an Automation Process

Planning, Design and Development

During this phase you create Automation strategy & plan, which contains following details-

- Automation tools selected
- Framework design and its features
- In-Scope and Out-of-scope items of automation
- Automation test bed preparation
- Schedule and Timeline of scripting and execution
- Deliverables of automation testing

Test Execution

Automation Scripts are executed during this phase. The scripts need input test data before there are set to run. Once executed they provide detailed test reports.

Execution can be performed using the automation tool directly or through the Test Management tool which will invoke the automation tool.

Example: Quality center is the Test Management tool which in turn it will invoke QTP for execution of automation scripts. Scripts can be executed in a single machine or a group of machines.

Maintenance

As new functionalities are added to the System under Test with successive cycles, Automation Scripts need to be added, reviewed and maintained for each release cycle. Maintenance becomes necessary to improve effectiveness of Automation Scripts.

Automation tools

Following are the most popular test tools

QTP: HP's Quick Test Professional (now known as HP Functional Test) is the market leader in Functional Testing Tool. The tool supports plethora of environments including SAP, Java, and Delphi amongst others. QTP can be used in conjunction with Quality Center which is a comprehensive Test Management Tool.

Rational Robot: It's is an IBM tool used to automate regression, functional and configuration tests for client server, e-commerce as well as ERP applications. It can be used with

Rational Test Manager which aided in Test Management Activities

Selenium: It's an open source Web Automation Tool. It supports all types of

web browsers. Despite being open source it's actively developed and supported

Benefits of automated testing

- 70% faster than the manual testing
- Wider test coverage of application features
- Reliable in results
- Ensure Consistency
- Saves Time and Cost
- Improves accuracy
- Human Intervention is not required while execution
- Increases Efficiency
- Re-usable test scripts
- Test Frequently and thoroughly
- More cycle of execution can be achieved through automation

CONCLUSION

Software testing is a critical element in the software development life cycle and has the potential to save time and money by identifying problems early and to improve customer satisfaction by delivering a more defect-free product.

Right selection of automation tool, testing process and team, are important players for automation to be successful. Manual and automation methods go hand-in hand for successful testing.

REFERENCES

1. Software Testing and Continuous Quality Improvement, by W. Lewis, et al (2008).

2. Experiences of Test Automation: Case Studies of Software Test Automation, by D. Graham et al (2012).

3. Implementing Automated Software Testing, by E. Dustin, et al (2009). Black-Box Testing, by B. Beizer (1995)

4. Software Testing: An ISTQB-ISEB Foundation Guide, by P. Morgan, et al (2010).