



Research Article

Utilizing Object Oriented Software Testing Tools in Microsoft Corporation

Jennifer Mosheshe*

Abstract

The paper seeks to access the utilization of object oriented software testing tools and its ability to improve the efficiency of new software production in Microsoft Corporation. In recent times, new software is becoming increasingly complex with a whole lot of risks associated with failure when being utilized. This has indicated that far greater importance has to be placed on testing to guarantee the software works properly as well as corresponds with the customer requirement. This document looks at software testing of new software being produced in Microsoft Corporation using several software testing tools and techniques that can in turn help improve the efficiency and cogency of the software testing process as well as the quality of new software that are produced in the company. The results show that object oriented software testing tools actually work on the new software produced by Microsoft Corporation. Basic problems that are needed to be addressed when applying various testing tools during testing process demonstrated the need and effectiveness for the utilization of object oriented testing tools. Furthermore, suggestions were made in respect to the necessary refinements to the existing ways in which Microsoft test their new software in respect of getting the best out of the organization.

Keywords

Object oriented software testing tools; Utilization; Microsoft corporation

Introduction

In the world today, information technology is becoming undeniably impossible to live without due to the fact that computer software are now being used for everyday use such as playing games, listening to music, preparation of office documents and so on. Also computer software are needed in almost every departments in various organizations such as accounting departments who use computer software CDs such as QuickBooks and Peachtree to manage their records and so many other departments who need computer software as well. This computer software cannot function properly without proper testing, which can be achieved with various software testing tools. Because of the fast growing dynamic needs of computer software in various organizations today, for any firm to survive, object oriented programming tools will be much needed while testing software in order to improve the quality of the organizations software programs as well as their products in general in order for the firm to achieve a

high standard and as well improve in the fast growing world. In this project, Microsoft Corporation is the organization that will be closely looked into in order to evaluate a selection of software testing tools that can be of great benefit to its new software that are being produced. Also, a list of software testing tools will be analyzed in details in order to emphasize their urgent needs to the organization. Firstly this project will be looking at the need of software testing in Microsoft Corporation and problems faced by most software companies that don't see testing as necessary. Also, different levels of testing will be emphasized, object oriented programming tools and their impact on how they can be used to improve software testing in Microsoft corporation will be explained, more detailed documentation on testing new software that have been completed in other satisfy user demand will be closely looked into and some clear examples will be highlighted. Finally we will also look at one way unified modeling language (UML) is used to automate the software testing process [1].

Project Aim and Objectives

This project is aimed at the importance of software testing with the use of object oriented testing tools in the software testing industry (Microsoft Corporation) and how it impacts new software in Microsoft Corporation. The dissertation will be looking at different levels of software testing tools, methods and techniques which take place during the software development life cycle. How object oriented software testing tools can improve the performance of software testing by taking advantage of some modeling languages like UML in order to swiftly develop test cases. Also how they help in reducing the testing time during the testing process and in turn produce improved new software in the Microsoft Corporation [2]. This research focuses on:

- The mapping of software testing tools and methods that are used in different departments in Microsoft Corporation.
- Identifying how these tools can bring about improvements in test practices that are used within Microsoft Corporation.
- Performing a pre-study on the new software produced by Microsoft Corporation, how the company operates and how improvements can be made through these testing tools.
- Ability to enable the transfer of knowledge of these test practices that are use within the different departments in Microsoft Corporation.
- Ability to provide suggestions on how to achieve the best result sing object oriented testing tools in terms of testing new software among various departments in Microsoft Corporation.
- To educate the company on how these software tools benefits their organization as a whole.

Microsoft Corporation Background

Microsoft Corporation is an international software company that produces new software into the world market. It develops, supports, licenses and manufactures a wide range of software components that is needed in the computing industries across the globe. It is one of the largest software producers in the whole world and has

*Corresponding author: Jennifer Mosheshe, Department of Information Technology, Cardiff Metropolitan University, Cardiff, United Kingdom, E-mail: jennymosh@yahoo.com

Received: June 19, 2017 Accepted: July 19, 2017 Published: July 24, 2017

been producing new software for over 30 years now. Microsoft Corporation was founded in the year 1975 by Bill Gate and Paul Allen and has produced different new software such as BASIC interpreters, personal computer operating system, Microsoft windows operating system, Skype technologies, PC operating system, office suite market, Microsoft surface tablet computers, video games industry such as Xbox One console and mobile phones. All these new software are upgraded concurrently as well as new models of this software are being developed regularly [3]. As big and powerful as Microsoft corporation it had experienced a lot of down side when it least expects such as in the year 1998 Windows 98 was being launched in an open event called the Microsoft courtesy program, the software was supposed to say “hey I plugged in a new device” and load some appropriate drivers when turned on but instead a blue screen of error messages showed up due to bugs in the system in front of a large crowd, this caused a whole lot of embarrassment for the company as well as for the developers and the testers in the company and also led to lower sales of the product in the market which led to financial loss of the company. It was tagged as the USB blue screen failure [3]. Also other number of times such recurrence has occurred in the company such as Windows phone voice recognition failure where a windows phone was being lunched and it failed to work properly by not recognizing the voice as designed, fake prerecorded Kinect presentation where a video game failed by Microsoft while being played, surface demo for windows store froze during a presentation and so on. All these could be avoided with the help of software testing [4].

An Insight into Software Testing

Software testing can be defined as the process of fixing the value of a software item in other to discover differences between the given input and the expected output of the software. It also assesses the distinctive attribute of a software item before releasing it out to the world market. It can also be termed as the verification and validation process due to the fact that it is usually carried out during the development process of a software life cycle [5]. Software testing has now become an increasingly very important part of the development of new software because failures of software have brought about poor image to the software industries in general. Due to this bad and poor image caused by the lack of software testing, together with the rapid rise in the use of computers in Business-critical and Safety-critical areas have forced millions of testers and developers to realize the great importance of software testing which has made them to place a high emphasis on software testing in other to be convinced that a particular software works properly and corresponds correctly with customer requirements before releasing it, in other to avoid disgrace [5]. Software testing has not been taken seriously or considered as an important aspect of IT and software development as a whole due to the fact that many industries viewed the process of fully testing a software as a waste of time and unnecessary expenses which made software developers give up their software for testing only at the end of the development process which caused a set back by just testing the software for launching and just making sure it worked instead of checking if it satisfies customer requirement and performing intensive testing and checks in other to avoid long term failure. The result of this was that a large quantity of software did not match user requirement as great amount of money and lots of times were exhausted in other to make changes to this software in other to correct the problems. Also these created a bad image for the software industries and created doubts in the heart of their customers, so that even after the problems have been corrected the software reliability was undermined when it was eventually delivered, due to the initial

delay from the testing lapses and their primary specification that is the purpose the software is being created for in the first place. This problem created a big financial loss in the industry [6].

Levels of Software Testing

The different levels of testing can be divided into functional and non-functional testing, below are a few:

Regression testing

This type of testing is done when there is a change to software in order to verify that bugs have not entered into the software and the functionality of the software has not changed. The main purpose of regression testing ensures that no bug fix which can cause another fault in the software is present. Its advantage is that speed is increased when the software is being marketed as there is great confidence that each area of the software is working properly [7].

Unit testing

This testing is performed by software developers before the whole setup will be handed over to the testing team in order for the testing team to officially execute the test cases. Its disadvantage is that the testing process is not able to catch the entire bug in the software because the test data and scenarios used by developers to verify the source codes are limited [7].

Integration testing

This type of testing encompasses testing combined parts of software in order to decide if they work correctly together. Integration testing has two methods namely, Bottom-up Integration testing where lower levels are tested first before higher level and top down Integration testing where higher levels are tested first before lower levels [8].

System testing

In this type of testing the whole software system is tested once all the components in the system have been integrated in other to ensure that the software meets up with the company standards. Its advantage is that the software is usually tested thoroughly which helps in verifying and validating user requirement and application architecture [7].

Acceptance testing

This type of testing is usually conducted by the Quality Assurance Team, whereby they have pre written scenarios and Test Cases which are used to test the software application. The main intention of acceptance testing is to measure if the software meets the intended user specifications in order to satisfy the user requirements [8].

Performance testing

This type of testing is used to identify performance issues such as delay in network, database transaction processing and data rendering. This type of testing is mandatory because speed and capacity of software is very important in the world market [8].

Security testing

This type of testing encompasses the testing of software in order to identify any faults that are in the software from a security point of view. It ensures that the software is available for users, has integrity, confidentiality, and is well authorized and authenticated [7].

Object Oriented Testing Tools

Object oriented testing tools are tools used to aid testing of object oriented programs in an organization. They invoke actions on various GUI components through various APIs platforms. An example is stimulating a mouse click on java component J Button. An object oriented programming tool will generate the keyboard and pointer events on the operating system level by posting to the system event queues and the verifies the test result by checking that the specific GUI component is in existence [9]. There are various types of object oriented testing tools, few are listed below:

T-VEC test generation system

It is a test tool that helps a developer decides what test is needed to be executed. It is integrated in the design environment of software testing in order to supports the management and development of object oriented requirement specification [10].

Vermont high-test plus

It is a test tool that is used to automate the execution of test for those products with graphical user interface (GUI). It is used in the GUI environment for object level record and playback of all the various Windows and VB control [11].

TOOTSIE

It is a test tool that is used to automate the execution of test for those products without graphical user interface (GUI). It is used in the management environment for total and complete object oriented testing support [10].

Object-tester

It is a tool that helps implement test by making failures obvious. It is used in the implementation environment for automating the generation of C++ unit test script [11].

Total-metric for Java

It is a test tool that helps testers evaluates the quality of test done. It is used in the evaluation environment in calculating and displaying objecting oriented metrics for java language [10].

Object-detail

It is a test tool that analyzes a program without running it. It is used in the static analysis environment in automating the metrics generation of C++ programs [12].

Microsoft Corporation Software Production Process

Microsoft Corporation goes through different phases when producing new software by firstly coming together as a team to identify what exactly the opportunities in the market are at present and then they determine the growth of the market entry in terms of fixed cost, volume and price/cost margin. After which they create a mission and vision statement, goals and objective for the software, they issue ownership and then educate the strategic plan to their team members. After this, they go into promotions, advertisement and making a lot of noise for the whole world to know that new software will soon in the market in order to raise the interest of their customers. After this the designing of the software begins, thereafter they schedule the software, implement the software, test and ship the software to different part of the world where the product can then be utilized. All these activities are carried out by a group of IT Specialists

in the company who are called team members [13]. During software development process in Microsoft Corporation, it has been discovered that not only the lack of software testing that can cause the failure or delay of new software. There are other factors such as various types of development models, such models also have some negative effects on new software. These can be minor effects but sometimes they can be major effects and can really affect software from working properly or from being produced swiftly. Below are various stages in software development:

Waterfall model

This is a type of model in the system development life cycle that is used during software production. It is a step by step process where the next phase gets input from the previous phase during software development. It is an iterative model and a little change in a previous stage can cause a whole lot of delay and problems in the next stages. This delay can affect software production negatively since the software can't be brought to the market at the expected time [14].

Prototype

This is a way of modeling the actual idea of new software. It is the approximation of the final software to be produced. It is also a very slow process as customers are too involved in it which can cause too much change in the software production and can also cause a delay and even change the whole specification of the initial software that was intended to be developed [15].

Agile

This is an incremental model where software is developed in an incremental swift circle. Due to the fact that it is a complex type of model it can be hard for developers to know the amount of effort to be put into the software development which can cause failure of the software at the end of the development process [14].

RAD

This is a quick type of development model that is incremental, where they are developed in parallel just like a mini project; it requires a very strong team because it is difficult to identify the business requirement in this model which can in turn cause a wrong requirement specification of the new software to be developed [16].

In the year 1970, typewriters, mimeograph and carbon paper were used for copying documents that were needed by organizations. After some few years Bill Gate and Paul Allen saw that this was a setback to the world at large and decided to come together with the hope of creating new software that can improve production in various organizations. They decided to start a computing organization called the Microsoft Corporation. The aim of the organization was to help organizations around the world achieve their full potentials by improving and creating new software to advance the course of various organizations [4]. In the year June 1975, decided to form a partnership since they have the same vision and named it Microsoft. They began by making desktop computers to be located in every home, offices and schools. Gradually changes in the way Microsoft operates where made. In the year 1980, a new operating system was created, this operating system is software that runs computer hardware and bridges the gap between these computer hardware and foundational programs where other computer programs can run. This operating system is called the Microsoft Disk Operating System [10].

In the year 1981, “C:” language was introduced and other cryptic commands such as the backslash (\) key was also introduced in order to improve speed in the computer world. In the year between 1982 and 1985, Microsoft Corporation introduced the Windows 1.0 which was actually the first version of their new software. Since then Microsoft Corporation have been developing new software, they moved from Windows 2.0 to 2.11 from 1987 to 1992 and then to Windows 3.0 from 1990 to 1994. From there they moved to Window NT in 1993, Windows 95 in 1995 to 2001, Windows Me, Windows 98 and Windows 2000 in 1998 to 2000. Then they moved on to Windows XP from 2001 to 2005 which was very stable, fast and user friendly. By 2009, they moved on to Windows 7 and 2012 they moved on to Windows 8 which is the new software at present [17].

Relevance of Object Oriented Testing Tools

Microsoft Corporation keeps producing new software for various purposes to keep up with the unsatisfied demands of the fast growing computer age in the world today. All of this software needs to be tested accurately in order to avoid failures and because most or all of this software are dynamic in nature, object oriented testing tools aid their testing process by making the processes easier, faster and more reliable. There are various high success rate activities that are performed using these object oriented software testing tools such as software debugging, Static Analysis, test Management, Version handling, documentations showing test cases and test reports, checking compliance with organizational standards, hardware in loop and many more [18]. These software testing tools are of several categories, some are for sale while others are open source testing tools which are for free and are available for the public. The basis of these tools is that when a developer want to build a new software based on the requirement of the user, the developer might end up working for very long hours without the help of proper testing, that is where the tester comes in so as to make sure the user gets exactly what he wants while also helping the developer achieve this goal in a short time. The use of these object oriented testing tools will help Microsoft Corporation increase productivity of new software at a fast pace [19].

Conclusion

Amongst all the various types of software testing tools, object oriented software testing tools are still reported to be the most robust and highest speed capacity during software testing process, which is also capable of improving both the organisation and actual software functions to the best capacity [20]. However, the current utilization measurability for object oriented testing tools in Microsoft Corporation for both internal and external software production process is not high enough. This could be due to problems during the system development life cycle such as the waterfall model, prototype, agile and RAD development process. Besides, subsidy by organisational management in improving the quality of testing tools for the reduction of software testing errors in Microsoft Corporation, the rate of testing errors are still relatively high. This study has adequately identified several factors that determine the utilization pattern of various testing tools in Microsoft Corporation. The project also suggests ways of improving the software testing process.

References

1. Berner S, Weber R, Keller RK (2005) Observations and lessons learned from automated testing, Proc. 27th Int Conf Soft Eng, 571-579, St. Louis, MO, USA.
2. Mark S, Philip L, Adrian T (2009) Research methods for business students. 5th edtn, Financial Times Prentice Hall, New Jersey, USA.

3. <http://www.microsoft.com/about/en-us/default.aspx>
4. <https://support.microsoft.com/en-us/products/windows>
5. <http://www.codeproject.com/Tips/351122/What-is-software-testing-What-are-the-different-ty>
6. <http://business.wfu.edu/default.aspx?id=262>
7. Biffi S, Aurum A, Boehm B, Erdogmus H, Grünbacher P (2005) Value-Based Software Engineering. Springer-Verlag, New York, Inc., Secaucus, NJ, USA.
8. Lionel CB, YvanL, Michal MS (2006) Automated, contract-based user testing of commercial-off-the-shelf components. 28th Int Conf Proc Soft Eng, Shanghai, China.
9. http://www.tplan.com/robot/docs/articles/img_based_testing.html
10. Pohjolainen P (2002) Software Testing Tools, Dissertation, University of Kuopio, Finland.
11. Nair VN, James DA, Erlich WK, Zevallos J (1998) A Statistical assessment of some software testing strategies and application of experimental design techniques. Statistica Sinica 8: 165-184.
12. Brownlie R, Prowse J, Phadke MS (1992) Robust testing of AT&T PMX/StarMail using OATS. AT&T technical J 71: 41-47.
13. Guthrie S (2017) The Microsoft Software Development Process.
14. Bertolino A (2002) ISSTA panel: is ISSTA research relevant to industrial users? ProcACM SIGSOFT Int SympSoft Test Analysis, Roma, Italy.
15. Bernat G, Gaundel MC, Marre B (2007) Software testing based on formal specifications. Software Eng J 6: 387-405.
16. Causevic A, Sundmark D, Punnekkat S (2010) An industry survey on contemporary aspect of software testing. 3rd Int Conf Software Test VerificatValidat (ICST), Paris, France.
17. Hawryszkiewicz I (2001) Introduction to system analysis and design. (5th Edtn), Prentice Hall, New Jersey, USA.
18. https://gupea.uu.se/bitstream/2077/29086/1/gupea_2077_29086_1.pdf
19. Dafydd V (2007) System testing with Oriented Program, Dissertation, Swansea University, London, United Kingdom.
20. Cohen DM, Dalal SR, Parelius J, Patton GC (1996) Thecombinatorial approach to automatic test generation. IEEE Software 13: 83-88.

Author Affiliation

Top

Department of Information Technology, Cardiff Metropolitan University, Cardiff, United Kingdom

Submit your next manuscript and get advantages of SciTechnol submissions

- ❖ 80 Journals
- ❖ 21 Day rapid review process
- ❖ 3000 Editorial team
- ❖ 5 Million readers
- ❖ More than 5000 
- ❖ Quality and quick review processing through Editorial Manager System

Submit your next manuscript at • www.scitechnol.com/submission