Software Testing Techniques to Make Application More Robust

Vineet Saxena
College of Computing Sciences and IT,
Teerthanker Mahaveer University, Moradabad, Uttar Pradesh, India

***Abstract:** The software companies have been divided in two categories as the product base and services base and both type of organisation have actively engaged in the development of the software that are used by the client and the customers . But the process of the development of the application is so long and involved many activities that there are high chance a that some error are remain in the application ,therefore it is necessary to resolved all the errors , defect or bugs , whatever it is, before delivered to the client finally . There are no such criteria that an application must be 100 percentages right as every application has some errors that cannot be resolved but the aim of the software organisation is to delivered a enough robust software that can be able to meet the requirements given by the client. The present paper has focused on the reliability of the test case for the satisfaction of clients as well as the customers.*

***Keywords:** Smoke testing, Sanity testing, software, Manual testing, developer, Application, Development.*

INTRODUCTION

The way of doing businesses around the world has been changing with a fast pace. The manual work of the worker has been replaced by the AI &ML methods. There many new businesses have evolved over the times that were not even imagining in the past. The some of the business, which could not be imagined in the past, are e-commerce and online booking of the transport instantly like OLA and Uber. Apart from these two examples, there are many more trade and commerce activity that are using the technology very deeply in their daily business activity. Apart from this, introduction of the smart phones based on the android have increased the use of the software based application for the mobile user. These installed applications in smart phones must be able to satisfy the customer need by functioning as per claim made in the specification. These applications are also the part of the businesses and a mean to generate the revenue for the companies who owned these applications.

It is evident that nobody will like a software application loaded with the errors and defects. The errors and defects in the application during the testing phase are known as the bug. A client needs a software based application for fulfilling the business need and satisfying the customer's requirement; therefore application must be bug free and will not fail or break during the operation [1]. Once an application is deployed on the production server, it is available for the customers in order to carry out the business activity. The smooth flow of the application without any flaws decided the satisfaction level of the customers.

The development of the software application is long process and many activities have been involved. The process started from the point where a project manager or business analyst gathers the information about the client's requirement and this is known the software requirement specification (SRS) as in figure 1. After gathering the requirement specification, a business analyst transformed it into an unapproved SRS and then technical architecture of the software firm made an approved SRS after consulting with the clients. The process of the SRS involved the feasibility study of the software to be developed as how many requirement of the client can be transform into the software and how many requirement cannot developed as the part of the software and what is the alternate solution of these requirements

SOFTWARE DEVELOPMENT LIFE CYCLE

The software company follow a model that is known as the software development life cycle (SDLC). The software organisation constitute a team comprises Project manager, business analyst, technical architecture, developers and testers. Each member of the team has their specific assigned task. The most important member of the team is the project manager who has the overall responsibility of the project. Whenever a client approached a software company for development of the software application, a team is constitute with the domain expert and a technical architecture and the Business analyst fix a meeting with the client for gathering the requirement of the client [2]. On the basis of the gathered requirement, Business analyst made an unapproved SRS.

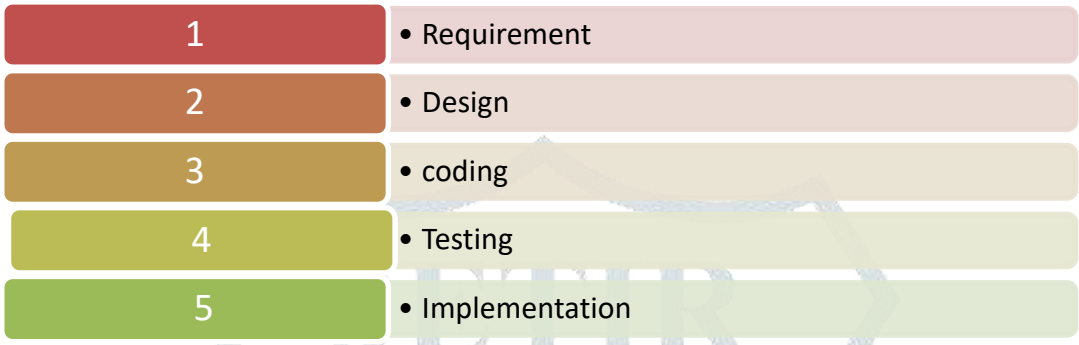

Fig. 1 Phases of Software Development Life Cycle

The unapproved SRS discusses with technical architecture along with the clients and check the feasibility of the various aspect of the requirement mentioned in the SRS. After detailed discussion, the unapproved SRS has turned in to the approved SRS. And then a walk through meeting has been called for the all team member, where business analyst and technical architecture gives a demonstration about the requirement of the application to be developed. These all process has been included into the requirement phase [3]. After the completion of the process of the requirement phase, the design phase has started, where technical architecture has prepared the mock up screen along with the designing of the other feature of the application.

TYPES OF SOFTWARE TESTING METHODOLOGY

The software testing Methodology can be divided into group as white box testing and black box testing. A third type of the testing is also existing known as the grey box testing (figure 2). The grey box testing is nothing but the combination of the black and white box testing. Whenever black box testing and white box testing is done by the same person or same team, testing is known as the grey box testing.

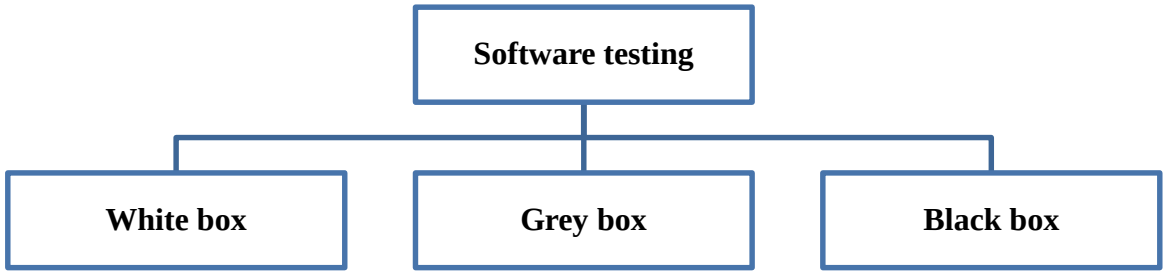

Fig. 2 Types of the Testing

White Box Testing:

White box testing is generally done by the developer or a team of the developers. Generally, in white box testing, the complete code has been checked and the issues resolved in every module or the unit of the software application. These types of the testing is known as the structural testing, transparent box testing and glass box testing as the developer can see the internal structural of the application i.e. the coding part of the application. As the tester rectify the problem at level of the coding done by the developer, therefore it is necessary that tester must have the knowledge of the coding process and it is more relevant if a senior member of the development team has done the testing over the application [4]. It enables the inner workings of the code to be examined; the specifics are not given any attention here. In order to generate desires, it is simply a method of supplying input values and controlling how the system processes. White Box Testing can be done at any level i.e. unit, integration or the system level. This type is efficient for detection and solving the problems as errors can be detected before they can cause any problem.

Fig. 3 Concept of Black Box Methodology

Black Box Testing:

Black box testing is the type of the testing where only the flow of the application is checked along with the client's requirement. Basically in white box testing, output of the test cases has been checked against the some input. The name black box testing derived from the fact that a tester does not need the coding knowledge for the black box testing and he does not need to see inside the box i.e. the coding part of the application. The black box testing is also known as the functional testing as it is done to check the functionality of an black box where some input are given and tester need to check the output (figure 3) The acceptance of the input data is the main criteria to ensure the free flow and bug free functionality of the application.

Grey Box Testing:

Grey box testing is the testing where a single individual or a single team completes the black box and white box testing. The most advantageous factor of the grey box testing is that same team handle the both type of the testing therefore the quality of the testing is good and the time consuming in the testing is reduced [5]. The functional testing and the structural testing features are combined in the grey testing.

CONCLUSION

The primary aim of any project is to produce any software product of the highest quality. That means that testing plays a very significant role in any software product development. Several research tools have been developed and are currently running on the market. In order to compete with other software developers and produce a top-quality product, learning about various software testing tools has become an important activity for the testing team. This paper discusses some significant and often used/high demand software and also presents a study on various automated testing tools used on various platforms and highlights the significance of automation testing over manual testing and concludes that

the hidden needs of the testing need to be understood and a software that confirms those hidden needs must be created.

REFERENCES

[1] Software Testing Fundamentals, "Integration Testing - Software Testing Fundamentals," *Software Testing Fundamentals*, 2011. .

[2] E. T. Barr, M. Harman, P. McMinn, M. Shahbaz, and S. Yoo, "The oracle problem in software testing: A survey," *IEEE Transactions on Software Engineering*, 2015, doi: 10.1109/TSE.2014.2372785.

[3] A. A. Sawant, P. H. Bari, and P. . Chawan, "Software Testing Techniques and Strategies," *Journal of Engineering Research & Applications*, 2012.

[4] G. Candea, S. Bucur, and C. Zamfir, "Automated software testing as a service," 2010, doi: 10.1145/1807128.1807153.

[5] S. Anand *et al.*, "An orchestrated survey of methodologies for automated software test case generation," *Journal of Systems and Software*, 2013, doi: 10.1016/j.jss.2013.02.061.