Research and Development on Software Testing Techniques and Tools

S

Tamilarasi T
VIT University, India

M. Prasanna
VIT University, India

INTRODUCTION

Software testing is the process of executing the program with the intent of finding the error. It is done to check if the system meets the requirements specified and be executed successfully in the intended environment. Testing checks the system whether it is "Fit for Purpose", also verifies whether the system does what it is expected to do. Software testing techniques are the different approaches and ways of ensuring that a software application is fully tested. Now days we can get lots of Software Testing Tools in the market. Selection of tools is totally based on the project requirements & commercial or free tools (Open Source Tools) you are interested. Off Course, free Testing Tools may have some limitation in the features list of the product, so it's totally based on what are you looking for & is that your requirement fulfil in free version or go for paid Software Testing Tools.

BACKGROUND

As software applications get more ever complex and intertwined and with the large number of different platforms and devices that need to be tested. Software testing techniques encompass everything from unit testing individual modules, integration testing an entire system to specialized forms of testing such as security and performance [1]. Testing is a process used to help identify the correctness, completeness and quality of developed computer software. With that in mind, testing can never completely establish the correctness of computer software. One definition of testing is "the process of questioning a product in order to evaluate it", where the "questions" are things the tester tries to do with the product, and the product answers with its behavior in reaction to the probing of the tester.

SOFTWARE TESTING TECHNIQUES

Testing Outlook

Testing outlook has an impact on the software testing process. Consider the following definition:

Testing is the process to prove that the software works correctly.

This definition sounds good, but the person who developed the software will only try to show that the software works correctly. This is the typical psychology of testing. The software will work correctly for the inputs that are given by that person which will obtain correct results. If some other input was given, the software will obtain wrong results which is not acceptable in a commercial environment.

Now consider this definition:

DOI: 10.4018/978-1-5225-2255-3.ch653

Testing is a process to prove that the software does not work.

If the aim of the test engineer is to prove that the software does not work, then the process can be considered as good. If the software performs well, then you can say that the software is very reliable. If the software works efficiently, after some days of testing, it does not mean that the software has no bugs at all.

So the definition would be:

Testing is the process to detect the defects and minimize the risk associated with the residual defects.

Verification and Validation

While going for testing, the two terms verification and validation have to be differentiated. Barry Boehm defines these terms as follows:

- **Verification:** "Are we Building the product right?"
- **Validation:** "Are we building the right product?"

Verification is done by the development team to ensure that the software is as per the specifications in the SRS document. It is to check whether the software conforms to specifications. Validation is to check whether the software meets the customer expectations. It is carried out with the involvement of the client.

Levels of Testing

Unit Testing

Unit testing aims at testing each of the components that a system is built upon. As long as each of them works as they are defined to, then the system as a whole has a better chance of working together. Where possible, all units that could possibly fail are tested at least one. Judgment is needed to decide what parts need testing. Some things such as accessors/mutators generally don't need to be tested. Components that are erroneous can be detected earlier. The scope of unit tests is even smaller that your traditional tests. If errors are detected, they are generally easier to fix.

Integration Testing

The purpose of integration testing is to verify the functional, performance, and reliability between the modules that are integrated. It is a software development process which program units are combined and tested as groups in multiple ways. In this context, a unit is defined as the smallest testable part of an application. Integration testing can expose problems with the interfaces among program components before trouble occurs in real-world program execution. Integration testing is component of Extreme Programming (XP), a pragmatic method of software development.

System Testing

- In system testing the behavior of whole system/product is tested as defined by the scope of the development project or product.
- It may include tests based on risks and/or requirement specifications, business process, use cases, or other high level descriptions of system behavior, interactions with the operating systems, and system resources.
- System testing is most often the final test to verify that the system to be delivered meets the specification and its purpose.
- System testing is carried out by specialists testers or independent testers.
- System testing should investigate both functional and non-functional requirements of the testing.