Buzzle.com ® *Intelligent Life on the Web*

Home   Topics   Featured Articles

Types of Software Testing

Software testing is a process of executing software in a controlled manner. When the end product is given to the client, it should work correctly according to the specifications and requirements stated by the client. Defect in software is the variance between the actual and expected results. There are different types of testing procedures, which when conducted, help to eliminate the defects from the program.

+4
Recommend

Testing is a process of gathering information by making observations and comparing them to expectations. - Dale Emery

In our day-to-day life, when we go out, shopping any product such as vegetable, clothes, pens, etc., we do check it before purchasing them for our satisfaction and to get maximum benefits. For example, when we intend to buy a pen, we test the pen before actually purchasing it, i.e. if it's writing, does it work in extreme climatic conditions, etc. So, be it the software, hardware, or any other product, testing turns to be mandatory.

Software testing is a process of verifying and validating whether the program is performing correctly with no bugs. It is the process of analyzing or operating software for the purpose of finding bugs. It also helps to identify the defects / flaws / errors that may appear in the application code, which needs to be fixed. Testing not only means fixing the bug in the code, but also to check whether the program is behaving according to the given specifications and testing strategies. There are various types of strategies such as white box testing, black box testing, gray box testing, etc.

Need for Software Testing Strategies

The types of software testing, depends upon different types of defects. For example:

- Functional testing is done to detect functional defects in a system.
- Performance testing is performed to detect defects when the system does not perform according to the specifications.
- Usability testing to detect usability defects in the system.
- Security testing is done to detect bugs/defects in the security of the system.

The list goes on as we move on towards different layers of testing.

Software Testing Methods

To determine the true functionality of the application being tested, test cases are designed to help the developers. Test cases provide you with the guidelines for going through the process of software testing, which includes two basic types, viz. Manual Scripted Testing and Automated Testing.

- Manual Scripted Testing: This is considered to be one of the oldest type, in which test cases are designed and reviewed by the team, before executing it.
- Automated Testing: This applies automation in the testing, which can be applied to various parts of a software process such as test case management, executing test cases, defect management, and reporting of the bugs/defects. The bug life cycle helps the tester in deciding how to log a bug and also guides the developer to decide on the priority of the bug depending upon the severity of logging it. Software testing to log a bug, explains the contents of a bug that is to be fixed. This can be done with the help of various bug tracking tools such as Bugzilla and defect tracking management tools like the Test Director.

Software Testing Types

Software testing life cycle is the process that explains the flow of the tests that are to be carried on each product. The V- Model i.e Verification and Validation Model is a perfect model which is used in the improvement of the software project. This model contains software development life cycle on one side and software testing life cycle on the other hand side. The checklists for software tester sets a baseline that guides him to carry on the day-to-day activities.

Black Box Testing: It explains the process of giving the input to the system and checking the output, without considering how the system generates the output. It is also known as Behavioral Testing.

Functional Testing: The software is tested for the functional requirements. This checks whether the application is behaving according to the specification.

Performance Testing: This testing checks whether the system is performing properly, according to the user's requirements.

Performance testing depends upon the Load and Stress Testing, that is internally or externally applied to the system.

1. *Load Testing*: In this type of performance testing, the system is raised beyond the limits in order to check the performance of the system when higher loads are applied.
2. *Stress Testing*: In this type of performance testing, the system is tested beyond the normal expectations or operational capacity.

Usability Testing: This is also known as 'Testing for User Friendliness'. It checks the ease of use of an application.

Regression Testing: Regression testing is one of the most important types of testing, which checks whether a small change in any component of the application affects the unchanged components or not. This is done by re-executing the previous versions of the application.

Smoke Testing: It is used to check the testability of the application, and is also called 'Build Verification Testing or Link Testing'. That means, it checks whether the application is ready for further testing and working, without dealing with the finer details.

Sanity Testing: Sanity testing checks for the behavior of the system. This is also called Narrow Regression Testing.

Parallel Testing: Parallel testing is done by comparing results from two different systems like old vs new or manual vs automated.

Recovery Testing: Recovery testing is very necessary to check how fast the system is able to recover against any hardware failure, catastrophic problems or any type of system crash.

Installation Testing: This type of software testing identifies the ways in which installation procedure leads to incorrect results.

Compatibility Testing: Compatibility testing determines if an application under supported configurations performs as expected, with various combinations of hardware and software packages.

Configuration Testing: This testing is done to test for compatibility issues. It determines minimal and optimal configuration of hardware and software, and determines the effect of adding or modifying resources such as memory, disk drives, and CPU.

Compliance Testing: This checks whether the system was developed in accordance with standards, procedures, and guidelines.

Error-Handling Testing: This determines the ability of the system to properly process erroneous transactions.

Manual-Support Testing: This type of software testing is an interface between people and application system.

Inter-Systems Testing: This method is an interface between two or more application systems.

Exploratory Testing: Exploratory testing is similar to ad-hoc testing, and is performed to explore the software features.

Volume Testing: This testing is done when huge amount of data is processed through the application.

Scenario Testing: Scenario testing provides a more realistic and meaningful combination of functions, rather than artificial combinations that are obtained through domain or combinatorial test design.

User Interface Testing: This type of testing is performed to check, how user-friendly the application is. The user should be able to use the application, without any assistance by the system personnel.

System Testing: This testing conducted on a complete, integrated system, to evaluate the system's compliance with the specified requirements. This is done to check if the system meets its functional and non-functional requirements and is also intended to test beyond the bounds defined in the software / hardware requirement specifications.

User Acceptance Testing: Acceptance testing is performed to verify that the product is acceptable to the customer and if it's fulfilling the specified requirements of that customer. This testing includes Alpha and Beta testing.

1. *Alpha Testing*: Alpha testing is performed at the developer's site by the customer in a closed environment. This is done after the system testing.
2. *Beta Testing*: This is done at the customer's site by the customer in the open environment. The presence of the developer, while performing these tests, is not mandatory. This is considered to be the last step in the software development life cycle as the product is almost ready.

White Box Testing: It is the process of giving the input to the system and checking, how the system processes the input to generate the output. It is mandatory for a tester to have the knowledge of the source code.

Unit Testing: Unit testing is done at the developer's site to check whether a particular piece / unit of code is working fine. It tests the unit of the program as a whole.

Static and Dynamic Analysis: In static analysis, it is required to go through the code in order to find out any possible defect in the code. Whereas, in dynamic analysis, the code is executed and analyzed for the output.

Statement Coverage: It assures that the code is executed in such a way that every statement of the application is executed at least once.

Decision Coverage: This helps in making decision by executing the application, at least once to judge whether it results in true or false.

Condition Coverage: In this type of software testing, each and every condition is executed by making it true and false, in each of the ways, at least once.

Path Coverage: Each and every path within the code is executed at least once to get a full path coverage, which is one of the important parts of the white box testing.

Integration Testing: Integration testing is performed when various modules are integrated with each other to form a sub-system or a system. This mostly focuses in the design and construction of the software architecture. This is further classified into Bottom-Up Integration and Top-Down Integration testing.

1. *Bottom-Up Integration Testing*: Here the lowest level components are tested first and then the testing of higher level components is done using '*Drivers*'. The entire process is repeated till the time all the higher level components are tested.
2. *Top-Down Integration Testing*: This is totally opposite to bottom-up approach, as it tests the top level modules and the branch of the modules are tested step by step using '*Stubs*', until the related module comes to an end.

Security Testing: Testing that confirms, how well a system protects itself against unauthorized internal or external, or willful damage of code; means security testing of the system. Security testing assures that the program is accessed by the authorized personnel only.

Mutation Testing: In mutation testing, the application is tested for the code that was modified after fixing a particular bug/defect.

These methods show you the output, and helps you check if the software satisfies the requirement of the customers. Software testing is indeed a vast subject and one can make a successful carrier in this field.

By Harshada Kekare
Last Updated: 9/20/2011

Like This Article?

| 4 | Like |

Follow:
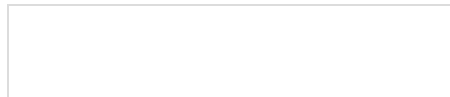
Related Articles

Software Testing Tutorial

Software Testing Interview Questions and Answers

Software Testing - Bug and Statuses Used During A Bug Life Cycle

Essential Steps of Software Testing

Manual Testing Tutorial

Manual Testing Interview Questions

Validation Testing

Post Comment | View Comments

Your Comments: