Buzzle.com® Intelligent Life on the Web

Search

Home Topics Featured Articles

Software Testing
Software Testing - Test Cases
Software Testing Interview Questions
Software Testing - Bug Life Cycles
How to Write a Software Test Plan
Manual Testing Tutorial

Software Testing Life Cycle

The software testing life cycle consists of various testing activities that need to be carried out to validate if the software meets the required design specification. It also explains which testing activity needs to be carried out and when. Different phases of the software testing life cycle have been explained in this article. Read on.

+2
Recommend

The software testing life cycle consists of a series of stages through which a software product goes through and describes the various activities pertaining to testing that are carried out on the product. Here's an explanation of the STLC along with a flowchart.

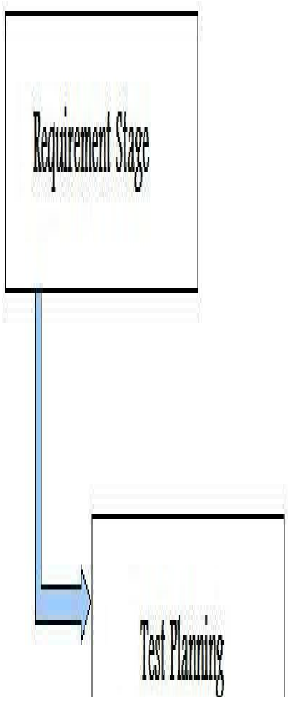
Introduction to Software Testing Life Cycle

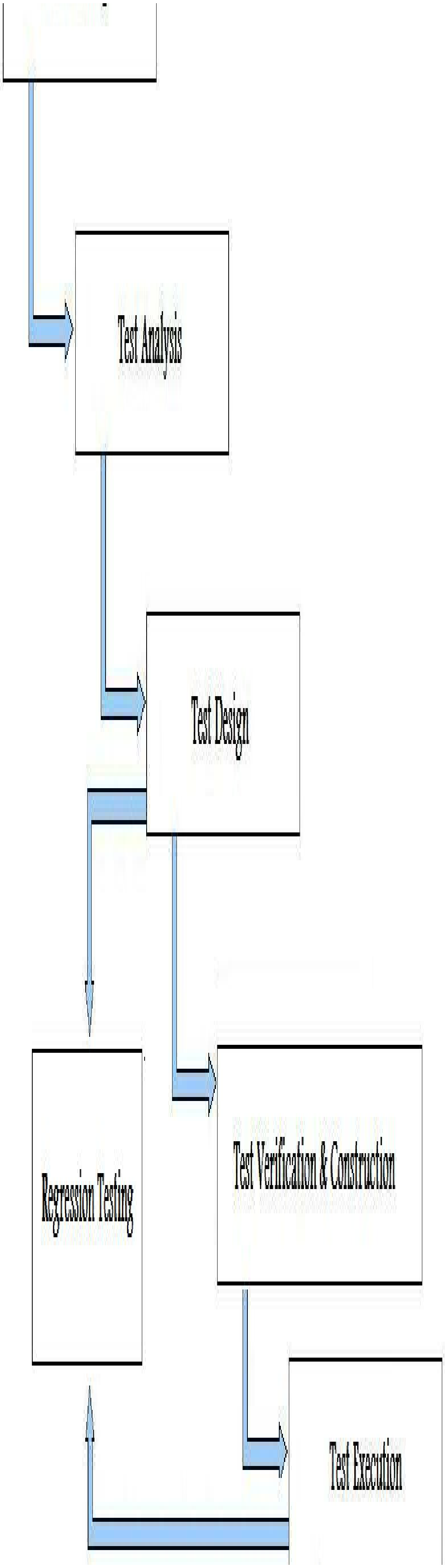
In every organization testing is an important phase in the development of a software product. However, the way it is carried out differs from one organization to another. It is advisable to carry out the testing process from the initial stages, with regard to the Software Development Life Cycle or SDLC to avoid any complications.

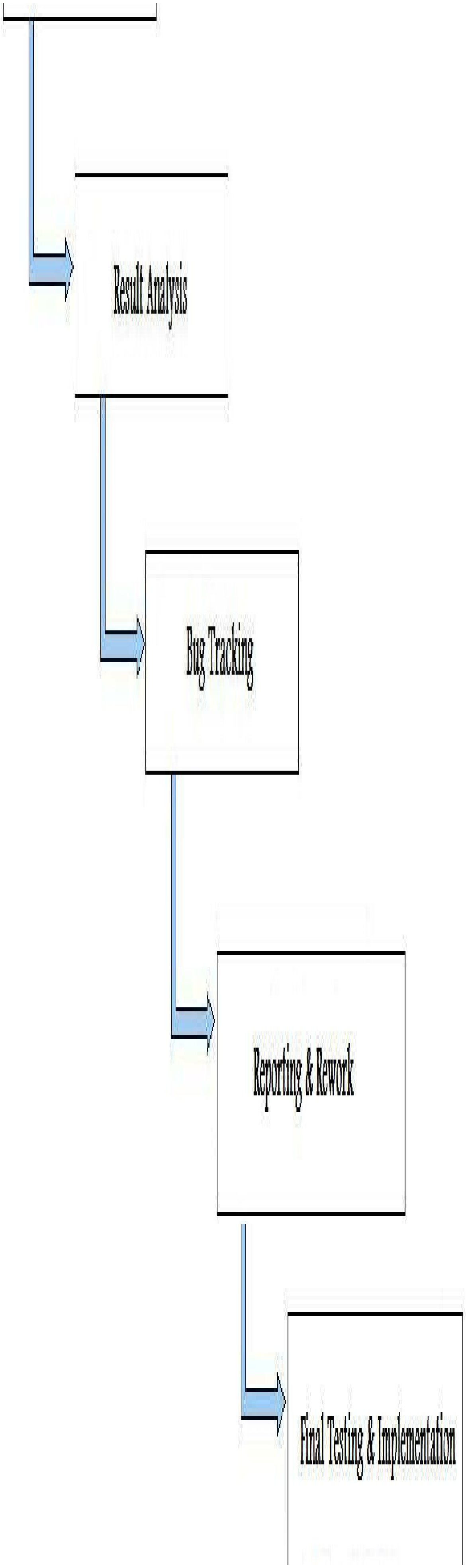
Software Testing Life Cycle Phases

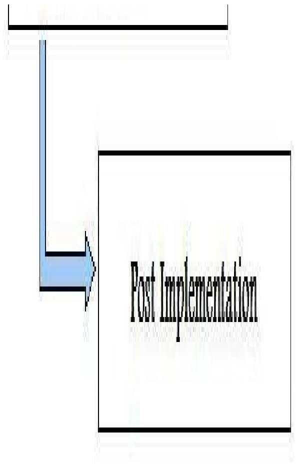
Software testing has its own life cycle that meets every stage of the SDLC. The software testing life cycle diagram can help one understand its various phases. They are:

1. Requirement Stage
2. Test Planning
3. Test Analysis
4. Test Design
5. Test Verification and Construction
6. Test Execution
7. Result Analysis
8. Bug Tracking
9. Reporting and Rework
10. Final Testing and Implementation
11. Post Implementation




Test Analysis
Test Design
Test Verification & Construction
Regression Testing
Test Execution


Result Analysis
Bug Tracking
Reporting & Rework
Final Testing & Implementation



Requirement Stage

This is the initial stage of the software testing life cycle process. In this phase the developers take part in analyzing the requirements for designing a product. The role of software testers is also necessary in this phase as they can think from the 'users' point of view which the developers may not. Thus a team of developers, testers and users can be formed, to analyze the requirements of the product. Formal meetings of the team can be held in order to document the requirements which can further be used as software requirements specification or SRS.

Test Planning

Test planning means to predetermine a plan well in advance to reduce further risks. A well-designed test plan document plays an important role in achieving a process-oriented approach. Once the requirements of the project are confirmed, a test plan is documented. The test plan structure is as follows:

1. *Introduction*: This describes the objective of the test plan.
2. *Test Items*: The items that are required to prepare this document will be listed here such as SRS, project plan.
3. *Features to be tested*: This describes the coverage area of the test plan, that is, the list of features to be tested; that are based on the implicit and explicit requirements from the customer.
4. *Features not to be tested*: The incorporated or comprised features that can be skipped from the testing phase are listed here. Features that are out of scope of testing, like incomplete modules or those on low severity, for example, GUI features that don't hamper the process can be included in the list.
5. *Approach*: This is the test strategy that should be appropriate to the level of the plan. It should be in acceptance with the higher and lower levels of the plan.
6. *Item pass/fail criteria*: Related to the show stopper issue. The criteria used has to explain which test item has passed or failed.
7. *Suspension criteria and resumption requirements*: The suspension criteria specifies the criteria that is to be used to suspend all or a portion of the testing activities, whereas resumption criteria specifies when testing can resume with the suspended portion.
8. *Test deliverable*: This includes a list of documents, reports, charts that are required to be presented to the stakeholders on a regular basis during the testing process and after its completion.
9. *Testing tasks*: This phase lists the testing tasks that need to be performed. This includes conducting the tests, evaluating the results and documenting them based on the test plan designed. This also helps users and testers to avoid incomplete

functions and prevent waste of resources.

10. *Environmental needs*: The special requirements of the test plan depending on the environment in which the application has to be designed are listed here.
11. *Responsibilities*: This phase assigns responsibilities to people who can be held responsible in case of a risk.
12. *Staffing and training needs*: Training on the application/system and on the testing tools to be used needs to be explained to the staff members who are responsible for the application.
13. *Risks and contingencies*: This emphasizes on the probable risks and various events that can occur and what can be done in such situations.
14. *Approval*: This decides who can approve the process as complete and allow the project to proceed to the next level that depends on the level of the plan.

Test Analysis

Once the test plan documentation is done, the next stage is to analyze what types of software testing should be carried out at the various stages of SDLC.

Test Design

Test design is done based on the requirements of the project documented in the SRS. This phase decides whether manual or automated testing is to be done. In automation testing, different paths for testing are to be identified first and writing of scripts has to be done if required. An end-to-end checklist that covers all the features of the project is necessary in the test design process.

Test Verification and Construction

In this phase, the test plan, test design and automated test script are completed. Stress and performance testing plans are also completed at this stage. When the development team is done with a unit of code, the testing team is required to help them in testing that unit and report any bug in the product, if found. Integration testing and bug reporting is done in this phase of software testing life cycle.

Test Execution

Planning and execution of various test cases is done in this phase. Once the unit testing is completed, the functionality of the tests is done in this phase. At first, top-level testing is done to find out the top-level failures and bugs are reported immediately to the development team to get the required workaround. Test reports have to be documented properly and the bugs have to be reported to the development team.

Result Analysis

After the successful execution of the test case, the testing team has to retest it to compare the expected values with the actual values, and declare the result as pass/fail.

Bug Tracking

This is one of the important stages as the Defect Profile Document (DPD) has to be updated for letting the developers know about the defect. Defect Profile Document contains the following

1. *Defect Id*: Unique identification of the Defect.
2. *Test Case Id*: Test case identification for that defect.
3. *Description*: Detailed description of the bug.
4. *Summary*: This field contains some keyword information about the bug, which can help in minimizing the number of records to be searched.
5. *Defect Submitted By*: Name of the tester who detected/reported the bug.
6. *Date of Submission*: Date at which the bug was detected and reported.
7. *Build No.*: Number of test runs required.
8. *Version No.*: The version information of the software application in which the bug was detected and fixed.
9. *Assigned To*: Name of the developer who is supposed to fix the bug.
10. *Severity*: Degree of severity of the defect.
11. *Priority*: Priority of fixing the bug.
12. *Status*: This field displays current status of the bug.

Reporting and Rework

Testing is an iterative process. The bug that is reported and fixed by the development team, has to undergo the testing process again to assure that the bug found has been resolved. Regression testing has to be done. Once the Quality Analyst assures that the product is ready, the software is released for production. Before release, the software has to undergo one more round of top-level testing. Thus testing is an ongoing process.

Final Testing and Implementation

This phase focuses on the remaining levels of testing, such as acceptance, load, stress, performance and recovery testing. The application needs to be verified under specified conditions with respect to the SRS. Various documents are updated and different matrices for testing are completed at this stage of the software testing life cycle.

Post Implementation

Once the test results are evaluated, the recording of errors that occurred during the various levels of the software testing life cycle, is done. Creating plans for improvement and enhancement is an ongoing process. This helps to prevent similar problems from occurring in the future projects. In short, planning for improvement of the testing process for future applications is done in this phase.

Read More on:

- Types of Software Testing
- Software Testing Methodologies
- Software Testing Tutorial

This was an overview of the software testing life cycle. Software defects are not always caused by coding errors. In fact, requirement gaps can also result in errors in an application. Though software testing can be carried out at any stage in the development phase, the process is mostly implemented after the requirements for an application have been specified and the coding has been completed.

By Harshada Kekare
Last Updated: 12/30/2011

Like This Article?

2 Like 11

Follow:

Related Articles

- Software Testing - How To Go About For Beginners
- Software Testing - White Box Testing Strategy
- Software Testing - Black Box Testing Strategy
- Software Testing - How To Log a Bug (Defect)
- Software Testing - Check Lists For Software Tester
- Software Testing - Compatibility Testing
- Software Testing Interview Questions and Answers
- Software Testing - Bug and Statuses Used During A Bug Life Cycle
- Software Verification & Validation Model - An Introduction

Post Comment | View Comments

Your Comments:

Your Name:

Submit

Advertising Info | About Buzzle | Privacy Policy