

Role of STLC in Software Development Life Cycle(SDLC):A Review

¹Gurmeet Kaur, ²Parminder Kaur

^{1,2}D.A.V.College(Lahore), Ambala City

¹gurmeet_g@rediffmail.com, ²parminder.kaur.pam@gmail.com

Abstract : This paper reveals the concept of STLC (Software testing life cycle) and its role in SDLC. SDLC is a structure imposed on the development of software product. Testing is an important part of software development. Testing should be started as early as possible to make it a part of a process of deciding requirements. In this paper we have explained various phases of software development and importance of testing in SDLC.

Keywords: SDLC, Testing, Bug, STLC, Software Reliability

I.INTRODUCTION

In the today's fast growing world of the Internet customer wants to get products to be implemented and updated faster than their competitors. Customer want more for software releases with new features to be implemented in short time frame (as early as possible), but they don't like to work with defected software. As next version of the product will releases in next few days, & gets only a couple of days of testing before it is shipped. So due to this short time frame or continuous releases the more bugs gets piled up into the products and which gets fixed in the next release. This model has its own problems. Releasing such software with so many bugs into it may affects the user experience which makes bad impact on quality impression of your company brand. They will remember about the delivered bad quality product, so there will be Importance of testing which makes vital role in SDLC. Testing helps to improve the quality, reliability & performance of the system with all check what all functions software supposed to do & also check that Software is not doing what he not supposed to do. There are major Importance of testing in the part of SDLC and it is better to introduce testing in the early stage of SDLC phases so it help to identify the defects in the early stage & try to avoid the bugs finding & get resolve in the last critical stage.

II.SOFTWARE TESTING

According to Myers-"Testing is the process of executing a program with the intent of finding errors." Before explaining the role of Software testing, we would like to explain different phases of Software Development Life Cycle (SDLC) process and Software Testing Life Cycle (STLC). The phases of SDLC are:

1. Requirement gathering/problem definition
2. Requirement Specification
3. Software Design
4. Coding
5. Software Testing and evaluation
6. Debugging
7. Software Maintenance

1. Requirement Gathering- In this phase the needs of the user are gathered and translated into a written set of requirements. These requirements do not include any technicalities according to the developer.

2. Requirement Specification- In this the user requirements are specified in the developer's terminology. It describes the requested behavior of a required system.

3. Software Design - It is the process of translating user requirements into a set of external interfaces. It includes both the problem solving and planning up solution for the software. The output of this phase will consists of the specification, which describe both WHAT (System will do) and HOW (it will work).

4. Coding – In this phase coding is done based on the design document for a module.

5. Software Testing – It includes software testing verification and validation of the system just build.

6. Debugging – It is not a part of testing domain. It is a separate process performed as a consequence of testing. The goal of this phase is the failure with the help of symptoms identified, locate the bugs and errors and finally correct it.

7. Software maintenance – In this software is being corrected or enhanced after delivery of the final product.

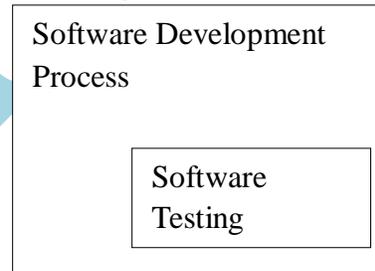


Fig.1 Testing process emerged out of development process

Four maintenance activities are-

- Adaptive Maintenance
- Perfective Maintenance
- Preventive Maintenance
- Corrective Maintenance

Software development is an engineering activity for a quality product, it consists of many processes, and Software testing has also emerged as a complete process in software engineering. Therefore our major concern in this paper to show that software testing is a process which runs parallel to SDLC as shown in fig-2.

Software Testing is a process that must be planned, specified, designed, implemented and quantified. Testing is a dual purpose process, as it is used to detect bugs as well as to establish confidence in the quality of software. The

testing process divided into a well defined sequence of steps is termed as STLC. STLC involves the testers at early stages of development. This has significant benefit in the project schedule and cost.

S/W Development Process	S/W Testing
Requirement gathering	
Requirement Specification	Test Plan
Design	Test Case Design
Code	Test Execution
.....
.....

Fig.2

STLC consists of the following phases.

1. Test Plan
2. Test Design
3. Test Execution
4. Post Execution/Test Review
- 5.

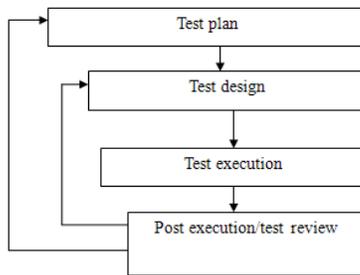


Fig 3

Test Plan: - The main goal of this phase is to consider the important issues of testing strategy like resources, schedules, responsibilities, risks and priorities as a roadmap. The output of this phase is test plan document. Test plans are developed every phase of SDLC.

Test Design: - In this phase Test Cases are designed. It includes the following activities.

- Determine the test objectives
- Preparation of list of items to be tested
- Identification of test cases.
- Selection of test case design techniques.
- Creating test cases and test data
- Setting up the test environment and supporting tools.
- Creating test procedure specification.

Test execution: - In this phase, all the test cases are executed including verification and validation. Verification test cases are started at the end of each phase of SDLC. Validation test cases are started after the completion of a

module. Test results are documented in the test incident reports, test logs, testing status and test summary reports.

Post Execution/ test review: - After the successful test execution, bugs will be reported to the concerned developers. This phase is to analyze bugs related issues and get feedback so that maximum number of bugs can be removed.

After fixing the bug, the developer reports to the testing team and modified portion of the software is tested once again. The final bug report are reviewed and analyzed for overall testing process.

As the project under consideration starts, testing too starts from the first level of SDLC, therefore the test strategy should be such that the testing process continues till the implementation of project as shown in the figure. The rule for development of a test strategy is that testing 'begins from the smallest unit and progresses to enlarge'. This means the testing strategy should start at the component level and finish at the integration of the entire system. To show the coordination between development process and testing various models exists but here we consider V model.

Although variants of the V-model exist, a common type of V-model uses four test levels, corresponding to the four development levels.

The four levels frequently used are

- Component(Unit)Testing
- IntegrationTesting
- SystemTesting
- Acceptance Testing

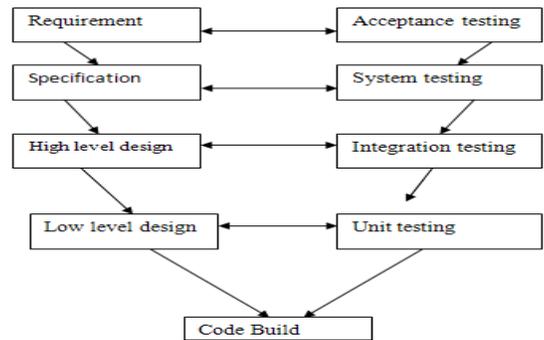


Fig. 4

Unit Testing: In this phase all the unit test case, created in the Low level design phase are executed. Unit testing is a white box testing technique, where a piece of code is written which invokes a method (or any other piece of code) to test whether the code snippet is giving the expected output or not. This testing is basically performed by the development team. In case of any anomaly, defects are logged and tracked.

Artifacts produced: Unit test execution results

Integration Testing: In this phase the integration test cases are executed which were created in the Architectural design phase. In case of any anomalies, defects are logged and tracked.

- Integration testing is a technique where the unit tested modules are integrated and tested whether the integrated modules are rendering the expected results. In simpler words, It validates whether the components of the application work together as expected.

Artifacts produced: Integration test results.

Systems testing: In this phase all the system test cases, functional test cases and nonfunctional test cases are executed. In other words, the actual and full fledge testing of the application takes place here. Defects are logged and tracked for its closure. Progress reporting is also a major part

in this phase. The traceability metrics are updated to check the coverage and risk mitigated.

Artifacts produced: Test results, Test logs, defect report, test summary report and updated traceability matrices.

User acceptance Testing: Acceptance testing is basically related to the business requirements testing. Here testing is done to validate that the business requirements are met in the user environment. Compatibility testing and sometimes nonfunctional testing (Load, stress and volume) testing are also done in this phase.

Artifacts produced: UAT results, Updated Business coverage matrices.

Comparative Study of SDLC and STLC

Phases	SDLC	STLC
Requirement and Information gathering	It is done by business development Team who analyze and files the complete analysis report.	Additional to the SDLC task, here the testing team adds into account any important prospect which may be additionally required.
Application Design	Architects works on low and high level design while the analysts work out the User Interface.	The Test team builds the overall design of the project based on complete logic and finally drafts the UI (user interface).
Coding drafting	Here coding is came out by the development Team.	Here the Testers or Testing teams prepares the test case.
Code Implementation	The dev team forwards the draft to the coders and they implement it to the project.	The final prepared Test case is put to test and implemented.
Trial and Testing	All kinds of testing aspects are tried and tested here.	All test cases and plans are put to test and reviewed for any bugs or errors.
Final Deployment	The project is pushed for deployment and finally enters the Market for End users.	Final testing and deployment is done here. Actually, it can be closely related to Trial and Testing phase but here the end users are professionals who will be finally using the software.
Maintenance and Support	Upgrades, Bug Fixes and Patches which are released in the due course of time.	Test case bottlenecks are modified and replaced with a more efficient and advanced module. Maintenance is also an added element here as well.

III. IMPORTANCE OF TESTING

Both system development process and testing are parallel to each other. STLC is a part of SDLC. It is like a set and subset; we cannot have STLC running individually on its own. STLC is the most important part of SDLC; one cannot release the final product without running it through STLC process.

Advantages of software testing in the Software Development Life Cycle:

- Testing should be introduced in the early stage of the SDLC, The cost of fixing the bug is larger if testing is not done in early stage & bugs found in later stages.
- In the today's competitive market only the quality product stays longtime firmly, so to make sure the produce the good quality product the testing of application is key factor in SDLC.
- As it not possible makes it software application is defect free but testing will be necessary.
- Most important thing of testing is the development environment is different than the Testing environment and the testing done on testing environment is similar to the Production environment.

So Testing play a vital role in the development process. Testing has a significant part in SDLC although the testing also upgrades the standards of the software and programmed

by recognizing errors prior in the system. It also upgrades the standard of organization.

IV. CONCLUSION

Software testing strategies and techniques is an important for improvement and measurement of a software and software development life cycle. It represents the review of specification, design and coding. Software testing also provides an objective, independent view of software to allow the business to appreciate and understand the risk at implementation of software. Various types of software strategies used for conventional and object oriented software development.

REFERENCES

- [1] <http://www.sdcl.ws/sdlc-and-stlc-complete-phase-by-phase-comparison/>
- [2] <http://www.softwaretestingstuff.com/2011/09/sdlc-vs-stlcdifferences-similarities.html>
- [3] "Software testing Principles and Practices" by Naresh Chauhan
- [4] "Software Engineering: A Practitioner's Approach" Roger S. Pressman
- [5] Involve testing throughout the SDLC available at <http://www.silverpath.com>
- [6] http://en.wikipedia.org/wiki/Software_testing