Interdependency and Segregation associates between software Development and software testing life cycles.

Dr. Maneesh Vilas Deshpande
Assistant Professor
Maneeshvdeshpande@rediffmail.com

Received: November 21, 2018 Accepted: December 27, 2018

ABSTRACT *Every Team Member Is Extremely Important In The SDLC. But The Most Important Players In Building Any Software System Are The Software Analysts, Software Developer And Software Tester. Because Of The Software Developer We Can Thing Of Or Rather To See The Goal But Due To The Software Tester We Can Actually/Nearly Reach To That Goal. The Main Focus Of This Work Is On The Software Development Life Cycle Model And The Software Testing Life Cycle Model, As Well As Their Interdependencies And Differences Between Software Development And Software Testing*

Keywords: *SDLC (Software Development Life Cycle), STLC (Software Testing Life Cycle), Software Analysts ,Software Developer , Software Tester.*

Introduction

In Any Development Phase We Have To Adopt Certain Steps , So That We Can Reach To The Intended Goal. SDLC Is Also Not An Exceptional Case. So In The Software Development Life Cycle We Also Follow Some Steps To Developed Customer Oriented Product. In SDLC Software Developer And Software Tester Are The Basic Building Blocks. But In Most Companies These Two Are Work Independently. Because Of Their Independent Work Many Problems Arises. But If They Work In Joint Venture Then This Software Development Is Called As Joint Application.[1]. A Software Development Process, Often Known As A Software Development Life Cycle (SDLC), Is A Framework That Governs The Creation Of Software. It's Frequently Thought Of As A Component Of The System Development Life Cycle. There Are Various Models For Such Processes, Each Of Which Describes How To Approach A Variety Of Actions That Occur Within The Process. ISO 12207 Is An ISO Standard That Describes How To Choose, Implement, And Monitor Software Throughout Its Life Cycle. Any Software Development Process Is Separated Into Logical Stages That Allow A Software Development Business To Efficiently Arrange Its Work In Order To Generate A Software Product With The Desired Capabilities Within A Given Time Frame And Budget. The Goal Of Any Software Development Project Is To Create High-Quality Software. [3]. The Software Development Life Cycle (SDLC) Can Be Defined In A Variety Of Ways, And There Are Several Alternative Models That Can Be Used To Complete It. At MKS (2009), The Organization Defines SDLC To Their Customers As "The Full Process Of Taking Formal, Logical Procedures To Create A Software Product" The Life Cycle, According To Scholars Hilmarsson, Oskarsson, And Yaqub (2005), Is "A Model On How To Bring A Concept To Reality," As Well As "A Systematic Manner For Developers To Produce A Working And Feasible System.. The SDLC Life Cycle Consists Of Several Phases Which Include Recognize The Customer Needs, Analysis, Design, Implementation, And Maintenance And Review Which All Play Important Roles In Supporting The Creation A Software Product. There Also Exists There Are A Variety Of Applications That Can Be Utilized To Help With The SDLC Process, Such As Prototyping Application, Rapid Application, Joint Application, And End-User Application.[1]. If We Try To Neglect Any Phase In SDLC Then We Can't Reach To Our Goal. This Study Is Organized As Follows: It Outlines The Interdependencies And Differences (Isolation) Between Software Development And Software Testing, And It Places Our Findings In The Context Of Previous Work (Literature Review). Following That, The Phases Of The Software Development Life Cycle Are Described, Followed By The Phases Of The Software Testing Life Cycle, And Lastly, The Conclusion And References Are Included.

Literature Review

The Software Development Life Cycle (SDLC) Can Be Defined In A Variety Of Ways, And There Are A

Number Of Alternative Models That Can Be Used To Complete It. SDLC Is Defined By MKS As "The Full Process Of Formal, Logical Procedures Taken To Produce A Software Product" For Its Customers. The Life Cycle, According To Scholars Hilmarsson, Oskarsson, And Yaqub, Is "A Model On How To Bring A Concept To Reality," As Well As "A Systematic Manner For Developers To Produce A Working And Feasible System." Moreover SDLC Can Be Involved In Several Applications Such As Prototyping, RAD, JAD And End-User Applications To Facilitate The Process [1]. Software Development Life Cycle Is The Most Important Element In Software Development. The Software Development Life Cycle (SDLC) Is A Technique That Shows The Complete Development Process And Should Be Used By A Software Development Company To Ensure A Successful Software Development. While Current SDLC Is Classified Into Two Primary Categories: Classic SDLC And Agile SDLC, And The Differences Between Them Are Explained. On The Basis Of Advantages And Disadvantages Between Above Two Methods We Can Say That It Is Critical That The Development Team Chooses An SDLC That Is Appropriate For The Project. Size Of Team, Geographical Position, Size And Complexity Of Software, Type Of Project, Business Strategy, Technical Capability, And Others Where It May Be Considered Acceptable Are Some Characteristics That Development Teams Could Utilize To Select The Ideal SDLC. A SDLC Selection And Adoption Process Is Critical Since It Ensures That An Organization's Chances Of Delivering Software Effectively Are Maximized, Therefore Choosing And Adopting The Correct SDLC Is A Must.

A Management Decision With Long-Term Consequences [2] Diverse SDLC Models, Such As Agile, RAD, And Waterfall, Are Utilized In Various Businesses Based On The Circumstances. For Example, The V-Model Provides Verification And Validation For Organizations And Is Highly Valuable. Each Of These Software Development Strategies Has Its Own Set Of Benefits And Drawbacks. [3] [5] Evolving A New Software Development Life Cycle Model (SDLC) That Includes Release Management. A Software Life Cycle, Dubbed The Y Model, Is Demonstrated In This Study, Among Other Things. The Y Model Outlines The Major Phases That Must Be Followed Under Its Cover.[6]

Interdependencies Between SDLC And STLC :

Software Testing Life Cycle (STLC) And Software Development Life Cycle (SDLC) Are Similar (ST LC) Are Given. Since SDLC And STLC Are Totally Different But Some Of The Key Aspects That We Considered Are Same Table 1 Shows Some Key Aspects Which Are As Follows. Considering All The Key Aspects One Must Say That SDLC And STLC Are Different But Finally Reach To That Same Goal That Is The Customer Satisfaction.

Key Aspects Consider	Software Development Life Cycle (SDLC)	Software Testing Life Cycle (STLC)
User Requirements	Essential	Essential
Development Direction	Important	Important
Testing	Required	Required
Customer Involvement	Necessary	Necessary
Implementation	Similar	Similar
ISO Standards	Follow	Follow
Expertise Required	High	High
Created Faulty Software	Affect Reputation Of Company	Affect Reputation Of Company
Intended Goal	The Customer Satisfaction	The Customer Satisfaction

Table.1 Interdependencies Between SDLC And STLC

Isolation Involved In SDLC And STLC:

Considering The Software Development Life Cycle (SDLC) And The Software Testing Life Cycle (STLC) Have A Lot In Common (STLC)

There Are Many Discrepancy Also Involved Between Two. Table 2 Shows Some Key Aspects Which Are As Follows.

Key Aspects Consider	Software Development Life Cycle (SDLC)	Software Testing Life Cycle (STLC)
User Requirements	Detail User Requirements	Iterative User Requirements
Rework Cost	High	Low
Development Direction	Fixed	Changeable
Testing	After Coding Phase Completed	Quite Earlier Than SDLC
Customer Involvement	Low	High
Extra Quality Required For Developer	Communication Skill, Interpersonal Skill	Nothing
Suitable Project Scale	Large-Scale	Comparatively Low-Scale
Maintenance	Required	Not Required

Budget	High	Low
Phases Involvement	Analyst, Designer, Developer, Tester	Tester, Team Leader, Team Manager
Orientation Of Technique	Process Oriented	Product Oriented
ISO Certification	ISO 12207	ISO 9126
Mentality Involved	Different Mentality	Same Mentality
Guarantee Of Success	Not Predictable	Predictable
Time Frame	Long	Medium
Code Checking	Performed	Not Performed
Automated Tools	Not Available	Available
Satisfaction Level	Sometime Less	More

Table. 2 Isolation Involved Between SDLC And STLC

Software Development Life Cycle Phases:

A Developer Of Software A Life Cycle Model Is A Set Of Actions With An Ordering Relationship Between Them That Are Executed In Such A Way That The Ordering Relationship Is Satisfied And The Intended Product Is Produced [6]. Every New Development Process We Have Certain Phases. SDLC Also Have Certain Phases And Which Are As Follows [See Fig. 1][7].

* Re-Cognization Of Need
* Study Of Possibility
* Research
* Planning
* Setup
* Follow-Up And Upkeep

Recognization Of Need: The First Phase, System Investigation Is Important Because It Is The Stage Where Problems And Opportunities Are Identified. This Stage Answers Questions Relating To The Worth Of The Problems Being Solved. This Phase Will Decide Whether The Project Is Going To Be Successful Or Not . There Are Many Ways To Collect Such Type Of Needs By Inspection, Walkthrough, Reviews Etc.

Feasibility Study : Output Of First Phase Is Taken As A Input To This Phase. On The Collected Data The Detail Study Takes Place To See The Needs Are Proper Or Not. If Needs Are Not Up To The Mark Then Once Again The Data Collection Procedure Takes Place At Much Deeper Level.

Analysis: This Phase Analysis Involves Examining Into Depth What The Designed System Must Accomplish. This Phase Produces A List Of Priorities And Requirements. This Phase's Goal Is To Identify A Document As Well As The Proposed System's User Requirements. This Is The Second Stage. Involves Interviews Some New Techniques To Find Out More Data Related To Desired System.

Design: The Outputs, Inputs, User Interfaces, Hardware, Software, Database, And Other Technical Requirements And Details That The System Must Meet Are Determined During The System Design Process.

Implementation : The Implementation Phase Is Where All Of These Decisions Take On A Physical Form. Installation And First Training Are Part Of This Phase, Which May Also Include Hardware And Network Upgrades.

Post Implementation And Maintenance : The Final Stage, System Maintenance And Review, Is Designed To Determine And Assure That The System Is Meeting Its Objectives.

This Phase Also Includes Making The Necessary Changes That Will Ensure Success

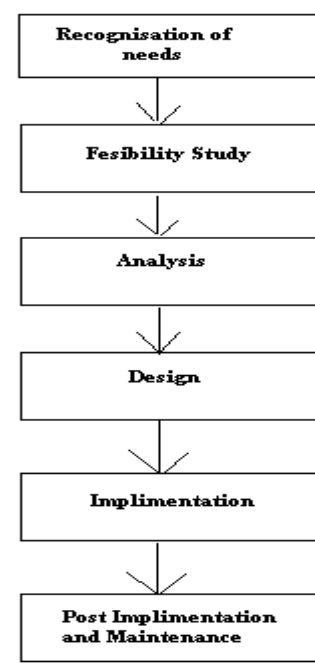

Fig.1 Software Development Life Cycle

Software Testing Life Cycle Phases:

A Software Testing Life Cycle Model Is A Collection Of Guidelines For Software Testing Of Activities Which Will Produce Desired Output. Following Are The Phases Which Is Introduced In The STLC.[See Fig.2] [8]

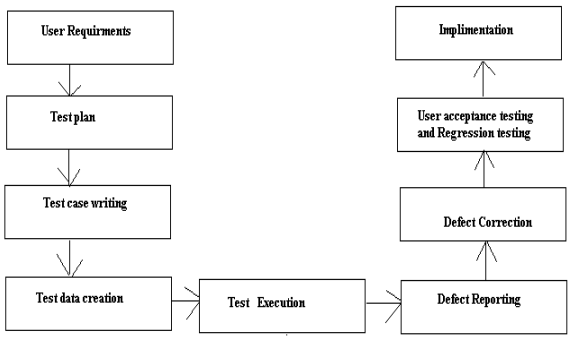

Fig.2 Software Testing Life Cycle

User Requirements: This Phase Is Similar To Re Cognization Of Study Used In SDLC. Where All The User Requirements Are Taken Through Specification Requirement Document This Phase Is So Important Therefore Called As Primary Specification Of STLC. This Phase Is Called As " All Or Nothing " Since If Requirements Are Simple And Not Changing Then We Reach To The Final Stage Quickly, But If The Requirements Are Changing Continuously And Are So Complex Then It Is Not Easy To Adopt This Changes. **Test Planning** : Planning Is The Next Step. In This Phase Number Of Test Cases, Team Members, Resources, Testing Tools, Hardware And Networking Strategy Etc Will Be Systematically Plan. A Test Bed Strategy Is Also Apply Here It Is A Testing Environment That Includes All Of The Hardware And Software Required To Test A Software Component Or System. This Is The Second Stage Also Include Identify The Reviews, Problem Reporting Procedure, Identify Problem Classification, Identify Acceptance Criteria, Identify Application Testing Database[9].

Test Case Writing: After The Work Is Assign To All The Team Members They Start Writing Test Cases[9]. Test Case Is Nothing But The Future Events That May Or May Not Be Happened During The Project. There Are Number Of Formats For Test Case Writing Used By Company. A Standard Format For Writing Test Case Is As Shown Below.

Tc_ID	Tc_Name	Objective	Steps	esting ata	Expected	Actul	Result
1)	Login Form (Valid)	To Verify That Login Form Is Open Or Not	Start-Program-Window-Intended Project	Used Mentioned Characters For User Name And Password	Login Form Should Be Open	Login Form Is Open	Pass
2)	Login Form (Invalid)	To Verify That Login Form Is Open Or Not	Start-Program-Window-Intended Project	Used Different Characters For User Name And Password	Login Form Should Not Be Open	Login Form Is Open	Fail

Fig.3 A Standard Test Case Format With An Example

For Example We Are Taken In To Consideration The Login Form Where Two Test Cases Are Written One For Valid Test Case And Another For Invalid Test Case Depends Upon The Valid And Invalid The Result May Be Pass And Fail.

Test Data Creation : In This All Possible Valid And Invalid Test Data Is Created By Tester To Execute The Test Case Of Previously Defined Test Case Format. There Are Number Of Test Cases Found In Any Project, If There Is Any Discrepancy Involved In The Project Then This Test Data Creation Format Is Forwarded To Customer[9].

Test Execution: By Applying Test Data All The Test Cases Are Executed To Test That They Meet Expected Result. If At All Any Deviation Between Expected And Actual Result Takes Place Then That Particular Test Will Fail.

Defect Reporting And Defect Correction : A Defect Is A Difference Between Expected And Actual Result /Output. In This Phase All The Failed Cases Are Reported As A Defect. While Reporting Of That Defect Is Assign To The Concern Developer ,Then Concern Developer Do The Needful Changes And Then Once Again Assign To Tester. If The Defect Is Sorted Then It Assign To Team Leader (TL) Or Team Manager (TM) This Phase Is Called As Defect Correction.

Regression And User Acceptance Testing: In Regression Testing Any Changes To The Application Should Not Introduced Unintended Behavior That Means Changes Should Not Affect Anything That Was Working Properly.

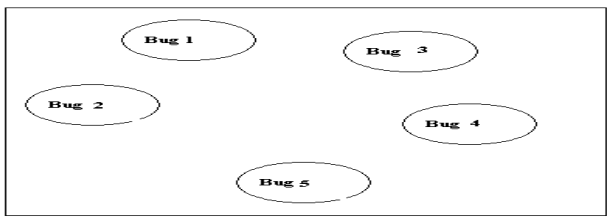

Fig.4 Regression Testing Example

In Figure 4 There Is A Regression Testing Example Shown. Where We Are Considered Five Different Types Of Bugs. And Also Note That These Five Bugs Are Not Dependent On One Another. In This Case If Developer In Concern With Tester Correct "Bug No. 1,3,5" Then All Remaining Two Different Kind Of Bug Will Not Affect The Already Corrected Bug. Where In User Acceptance Testing Test Conducted By N User At Customer Location In A Live Atmosphere To Find Out The Code Is Properly Working Or Not. Useful For Making Final Release Of The Product Acceptable By Clients Procedure Takes Place.

Implementation: This Is The Final Step Used In STLC. This Phase Is Similar As Used In SDLC. This Phase Is Nothing But The Stage Role Out Of A New Software. This Involves Installation And Initial Training And May Involve Hardware And Network Up Gradation.

Conclusion:

Both The Software Development Life Cycle (SDLC) And The Software Testing Life Cycle (STLC) Are Step-By-Step Processes. The User Requirements Feasibility Study, Analysis, Design, Implementation, Post-Implementation, And Maintenance Phases Are All Part Of The SDLC. Similarly, STLC Is Experiencing Some Difficulties Phase Starting From User Requirements, Test Plan, Test Case Writing, Test Data Creation, Test Execution, Defect Reporting, Defect Correction, Regression Testing, User Acceptance Testing, And Finally Implementation. Considering All The Phases Of SDLC And STLC We Come To The Conclusion That, Maintenance Phase Is Available In Only SDLC But It Is Not Appear In STLC. Since Testing Strategy Actually Works In Maintenance Phase. In This Paper We Described All The Phases And Some Examples Of Related Portion. Now The Main Aim Of That Paper Is To Relate The Two Terminology That Is SDLC And STLC. We Discuss Some Interdependency And Isolated Portion Where These Two Terminology Are Somewhat Different In Some Context. But Keep In Mind The Thing That SDLC And STLC Are The Most Important Aspects Of Any Development For Successful Completion Of Project.

ACKNOWLEDGMENT :

I'd Want To Express My Gratitude To All Anonymous Reviewers For Their Insightful Remarks, Which Were Used To Improve This Paper. I Am Grateful To Dr. Suryakant B. Thorat And Dr. Pradeep K. Butey For Providing Me The Appropriate Guidance And Assistance. Finally, But Certainly Not Least, My Family, All Of My Mentors Since Childhood, And My Friends For Their Unwavering Support.

References :

1) **Scholarly Articles** "Short essay on software development life cycle"2010 ss
2) **Yu Beng Leau , Wooi Khong Loo, Wai Yip Tham and Soo Fun Tan"**Software Development Life Cycle AGILE vs Traditional Approaches**"** 2012
3) **Sanjana Taya, Shaveta Gupta**"Comparative Analysis of Software Development Life Cycle Models"2011**.**
4) Software Development Life Cycle (SDLC) – the five common principles.htm
5) **Vishwas Massey, K. J. Satao** Evolving a New Software Development Life Cycle Model (SDLC) incorporated with Release Management International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-1, Issue-4, April 2012
6) **Luiz Fernando Capretz**Y: A New Component-Based Software Life Cycle Model 2005
7) **Carlo Ghezzi, Mehdi Jazayeri, Dino Mandrioli** "Fundamentals of Software Engineering 2nd Edition".
8) Software Testing Manual
9) **SANTOSH KUMAR SWAIN, SUBHENDU KUMAR PANI, DURGA PRASAD MOHAPATRA,** Journal of Theoretical and Applied Information Technology© 2005 - 2010 JATIT, "MODEL BASED OBJECT-ORIENTED SOFTWARE TESTING".